Full Circle Magazine

Issue #96

Get *unlimited access* to a cutting-edge technology and business library with **Apress Access!**

For $199

YOU GET:

- Unlimited access to Apress titles for a full year
- Instant access to each new Apress publication
- Compatibility with any device—desktop, laptop, or mobile
- Use of our new exclusive-to-Apress reader with unparalleled search functions
- Option to download any eBook for just $4.99 for a limited time

www.apress.com | @apress

Want more info? Check out www.apress.com/subscription

Apress ACCESS

The articles contained in this magazine are released under the Creative Commons Attribution-Share Alike 3.0 Unported license. This means you can adapt, copy, distribute and transmit the articles but only under the following conditions: you must attribute the work to the original author in some way (at least a name, email or URL) and to this magazine by name ('Full Circle Magazine') and the URL www.fullcirclemagazine.org (but not attribute the article(s) in any way that suggests that they endorse you or your use of the work). If you alter, transform, or build upon this work, you must distribute the resulting work under the same, similar or a compatible license. Full Circle magazine is entirely independent of Canonical, the sponsor of the Ubuntu projects, and the views and opinions in the magazine should in no way be assumed to have Canonical endorsement.

Please note: articles in this magazine are provided with absolutely no warranty whatsoever; neither the contributors nor Full Circle Magazine accept any responsibility or liability for loss or damage resulting from readers choosing to apply this content to theirs or others computers and equipment.

Welcome to another issue of Full Circle.

We have a crowded house for you this month. We have the usual suspects of Python and LibreOffice with a second part to last month's LaTeX article. Not only that, but I've added a fourth HowTo on Programming in JavaScript. That's the start of a new series. It doesn't end there. Heavens no. There's a new column covering Chrome OS, and one covering Ubuntu phones. The Ubuntu phone column will mostly cover news and updates to that side of things, but to start it off is an interview with the VP of Mobile who is answering some of your questions that I was unable to answer in last month's review of the BQ Aquaris E4.5.

If you like the idea of Dropbox (et al) but not the idea of them holding on to your data then you should have a look at Charles' article this month on Owncloud. Think of it as Dropbox running on your own machine. You host your own data, but can still access it from afar.

For a bit of fun I've put up a quick, short, survey simply asking which *buntu flavor and release you love, or hate the most. It's *buntu flavors only. No Mint or derivatives. I'll leave it up for a couple of months and print the results in FCM#100. The URL is dotted throughout this month's issue.

Also, as I posted on the FCM Twitter, Facebook and Google+ accounts, this issue marks eight years of Full Circle. Eight years! I want to take this opportunity to thank everyone who's ever done anything for FCM. I certainly couldn't do this myself.

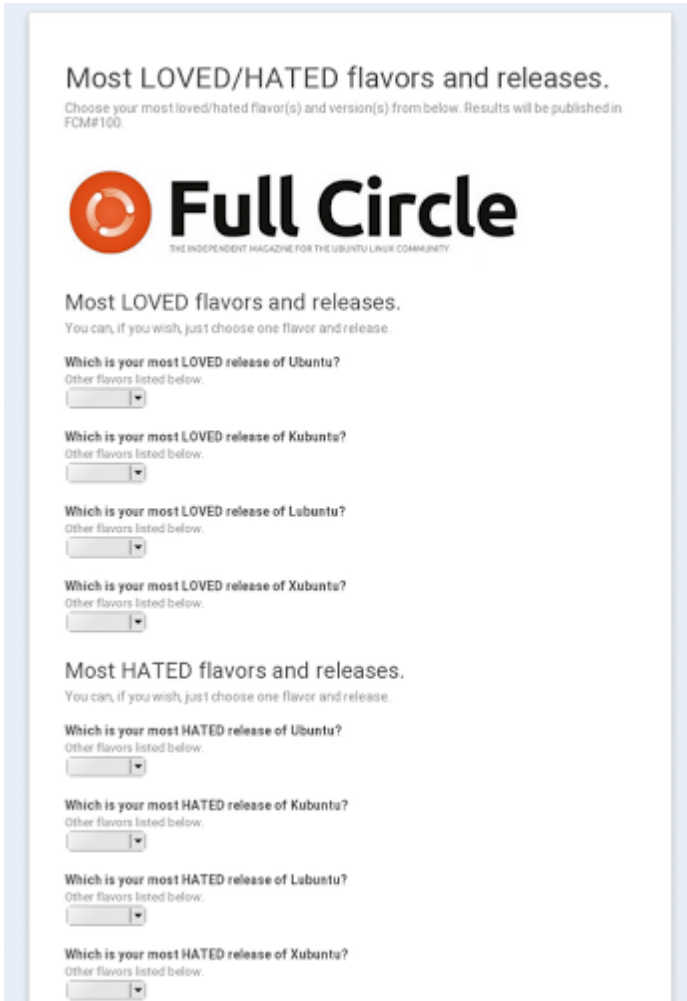
All the best, and keep in touch!

Ronnie

ronnie@fullcirclemagazine.org

Survey

You can find the survey here



Most LOVED/HATED flavors and releases.

Choose your most loved/hated flavor(s) and version(s) from below. Results will be published in FCM#100.

Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

Most LOVED flavors and releases.

You can, if you wish, just choose one flavor and release.

Which is your most LOVED release of Ubuntu?

Other flavors listed below.

Which is your most LOVED release of Kubuntu?

Other flavors listed below.

Which is your most LOVED release of Lubuntu?

Other flavors listed below.

Which is your most LOVED release of Xubuntu?

Other flavors listed below.

Most HATED flavors and releases.

You can, if you wish, just choose one flavor and release.

Which is your most HATED release of Ubuntu?

Other flavors listed below.

Which is your most HATED release of Kubuntu?

Other flavors listed below.

Which is your most HATED release of Lubuntu?

Other flavors listed below.

Which is your most HATED release of Xubuntu?

Other flavors listed below.

News

Ubuntu 15.04 Desktop, Phone and IoT: a converged world

Desktop

Ubuntu is the favourite environment for Linux developers, celebrated in products such as the Dell XPS 13 Developer Edition. This release introduces new tools for cloud and IoT development as well as making the desktop more productive with integrated menus and dashboard usability improvements.

Since 2008, Ubuntu desktop has been pre-installed on 40 million PCs worldwide and continues to grow in popularity as one of the world's most easy-to-use and reliable operating systems available today. Ubuntu-preloaded devices are growing in popularity in regions such as EMEA, India, South East Asia and Latin America, as well as being the high-end developer platform of choice in the West, with incredible machines such as the NVidia

DevBox for advanced machine learning and visualization engineering.

One of the key highlights in this release is the way in which developers are afforded immediate access to the most exciting and innovative software for their development on cloud, mobile and web.

Ubuntu Developer Tools Centre has become Ubuntu Make. With it, developers of mobile apps, web applications and games, have support for 14 new platforms, including:

- Android NDK & Android Studio updated to the latest release
- Other new IDEs: IDEA (ultimate and community editions), pycharm (professional, educational and community editions), webstorm, rubymine, phpstorm and eclipse
- Firefox developer edition
- Stencyl game development platform

These new features are also available to current 14.04 LTS users. Ubuntu Make provides a fresh, well-isolated development environment for every application, so developers don't have to worry about messing up up their installation if they want to install pipy, npm, rubygem libraries. Ubuntu's leadership in containers and containment enable fantastic developer experiences on modern workstations and laptops.

LibreOffice 4.4 brings increased business productivity, including the addition of improved change tracking to Writer, enhanced mail merge performance, and improved shapes which can now have fully formatted content with tables. Support for digitally signed PDF exports is a new addition with 15.04, as is support for connecting to Sharepoint and OneDrive. Some other updates of note

include:

- Addition of Statistics functions in Calc and password protected documents in Impress
- Improved OpenGL
- Support for slide transitions in Impress and Draw

Ubuntu Kylin 15.04

For Ubuntu Kylin users, locally integrated menus and click-to-minimise are now incorporated alongside upgrades to features applications such as Ubuntu Kylin Software Center, Youker Assistant, Youker Weather to name a few.

Youker Assistant V2.0.2 features a new user interface, allows better management of personal applications, and enables users to contribute directly to translation of package descriptions directly from the Ubuntu Software Centre.

Pinyin users can install vocabularies directly from the Sogou website.

Internet of Things: Snappy on Devices

Ubuntu Core is the smallest and most secure edition of Ubuntu. It is a super-lean, transactionally updated version of Ubuntu, perfect for inventors, technologists, and the active and growing Ubuntu developer community, for cloud container hosts, and smart, connected devices. It powers drones, robots, network switches, mobile base stations, industrial gateways, IoT home hubs.

“This is an amazing platform for the new generation of cloud and device developers” said Dustin Kirkland, product manager for Ubuntu Core. “The combination of an open platform with an app store that works across a very wide range of devices, from tiny embedded boards to

high end switches and routers, has stimulated a rush of creativity in the maker and entrepreneur communities."

The first stable version of 'Snappy' Ubuntu Core is now available as part of 15.04.

This release supports 64-bit Intel-based architectures, as well as ARM HF, providing a crisp common platform for device development that supports a very wide range of production hardware.

Ubuntu Phone

Ubuntu 15.04 is the base for the Ubuntu Phone currently shipping on the BQ Aquaris E4.5 and, in due course, on the Meizu MX4. The latest 15.04 updates will be delivered OTA (over-the-air) in coming weeks meeting Canonical's commitment to the mobile as an extension of Ubuntu on the desktop and cloud: a secure platform that is always up-to-date.

Ubuntu 15.04 will be available to download from http://www.ubuntu.com/download from Thursday 23 April.

Ubuntu 15.04: OpenStack ready to roll with Kilo and first outing of Snappy Core

This release delivers the new, snappy Ubuntu Core for transactional systems, such as cloud container hosts, smart devices, and a new container-based hypervisor, LXD, which sets a new benchmark for density and performance. With updated developer tools and the latest frameworks, languages, databases and packages, this is a significant release for Ubuntu professionals and developers.

Container-based hypervisor LXD sets records for density and speed

LXD, the next-generation hypervisor for containers, is now available in Ubuntu 15.04. LXD provides the full experience of virtual machines, the security of a hypervisor, and bare-metal performance and density.

"LXD eliminates the very high virtualisation penalty of traditional hypervisors, making Linux-on-Linux workloads much faster and much more dense," said Mark Shuttleworth. "Containers are the new frontier in virtualisation and cloud. We are delighted to lead with LXD and the integration of containers into OpenStack."

Early adopters include institutions with many Linux virtual machines running common code such as Tomcat applications under low load. LXD offers much higher density than KVM in these situations as the underlying hypervisor can consolidate common processes more efficiently.

In addition, workloads which are traditionally run on bare metal, such as Hadoop, perform at native speeds under LXD without the 15-20% overhead of KVM.

"LXD support in OpenStack means big data specialists can now use OpenStack APIs for provisioning, and get bare metal performance for their analytics," said Mark Baker, product manager for OpenStack in Ubuntu.

LXD provides a full "virtual machine" experience inside where administrators can run tools like Docker.

"LXD and Docker work together. LXD provides a full system container, like a virtual machine, and Docker provides the process container for applications," said Mr Baker.

This new release of Ubuntu provides:

- Live migration of LXD containers between hosts
- Automatic, rigorous security for containers through

kernel AppArmor profiles
- Docker 1.5

Snappy Ubuntu Core - Transactional Host for Containers and Things

'Snappy' Ubuntu Core is the new, transactional version of Ubuntu designed for lightweight cloud container hosts running docker and for smart devices. It contains all the familiar code and updates of Ubuntu, but is packaged with the new 'snappy' system, enabling guaranteed updates with rollback for both the OS and applications installed on it.

Ubuntu Core is the smallest Ubuntu available, designed for security and efficiency in devices or in the cloud.

"Ubuntu Core offers everything developers love about Ubuntu together with transactional updates," says Dustin Kirkland, product manager for Ubuntu Server at Canonical. *"Snap packages deliver apps securely to devices and cloud hosts, with isolation of application data and the guarantee that an update can be rolled back."*

This first version of Ubuntu Core features secure app containment and Docker 1.5, and is available on major public clouds and for ARM and x86 devices on a range of popular boards for IoT. Ubuntu Core is already running on the next-generation of network switches, home routers, smart drones and robots.

Ubuntu delivers OpenStack Kilo first

The Ubuntu OpenStack distribution from Canonical continues to be the most popular way to consume the latest OpenStack, with 64% of production OpenStack users choosing Ubuntu.

Telcos such as AT&T, NTT and Deutsche Telekom, large

enterprises such as Walmart and Best Buy, and service providers such as OVH, have adopted Ubuntu OpenStack as their preferred platform for private cloud. Canonical's professional services teams work with them to achieve the highest levels of scalability and efficiency from OpenStack.

Ubuntu OpenStack includes the widest ecosystem of certified software-defined network and software-defined storage from a partner list that includes Juniper, Cisco, Nuage, Plumgrid, MidoNet and many other vendors.

Canonical leads OpenStack development in key areas, notably container-based hypervisors (LXD) for performance and techniques for cloud scalability. In this release, the company has integrated ZeroMQ (0MQ) as a brokerless messaging system to eliminate bottlenecks in messaging as the cloud infrastructure scales. ZeroMQ is an intelligent transport layer for distributed applications. Canonical has pioneered the use of 0MQ to enhance the scalability of the core infrastructure itself.

Ubuntu will be the world's first OpenStack distribution to make the newest 'Kilo' release available to users, a significant step forward in scalability for virtual networks on OpenStack.

In Kilo, Neutron is updated to include Distributed Virtual Routing (DVR) to enable Neutron to scale more efficiently, and a preview of "DNS as a service" from the new 'Designate' component.

Cloud federation also takes a big step forward in Kilo with Ubuntu OpenStack now able to share identity across cloud regions. This enables enterprises with multiple OpenStack implementations to manage identity much more efficiently, and simplifies the path to hybrid cloud computing with OpenStack on premise and public

OpenStack clouds.

Canonical's OpenStack Interoperability Lab (OIL) continues to set the pace for multi-vendor interoperability, certification and testing of OpenStack. Canonical tests quality and interoperability on more than 3,000 cloud configurations each month, with 33 individual OIL partners on Ubuntu OpenStack, including Quanta, MetaSwitch, Nuage Networks, PMC Sierra, and Coho Data. Participation in OIL provides customers with confidence that components of an OpenStack cloud interoperate well together – making it easier to deploy and consume cloud resources.

Ubuntu on Public Clouds

Canonical's Certified Public Cloud (CPC) program includes the world's largest cloud providers such as Amazon AWS, Microsoft Azure and Google Compute Engine, and guarantees the Ubuntu experience for developers and users. It has expanded to include Scaleway, BrightBox and CloudSigma who gain a full portfolio of certified Ubuntu images and workloads. All Certified Public Clouds will be receiving images for Ubuntu 15.04 in the traditional server format together with new, snappy editions.

Ubuntu Server 15.04 is available for download at www.ubuntu.com/download from 23 April 2015.

Source: **Canonical**

Ubuntu on the Asus Zenbook UX305 ultrabook

The Asus Zenbook UX305 is a thin and light laptop that offers a pretty great value. For $699, you get a 2.6 pounds notebook with 8GB of RAM, 256GB of solid state storage, a 13.3 inch full HD matte display, and an Intel Core M

Broadwell processor.

For the software, you also get Windows 8.1 software but what if you'd rather run Ubuntu? The answer is absolutely yes, you can do that.

The operating system loaded quickly, and most of the hardware worked without any problems. I was able to connect to my WiFi network, surf the web in Firefox, watch YouTube videos, and install apps using the Ubuntu Software Center, among other things.

I did all of this while running from the Live USB image, but you should also be able to install Ubuntu to internal storage to either dual-boot Windows and Ubuntu, or to replace Windows altogether.

Source: http://liliputing.com/2015/04/ubuntu-on-the-asus-zenbook-ux305-ultrabook.html

Submitted by: **Brad Linder**

How open source software builds strong roots for better governance

'Open data' and 'going digital' are subjects high on the international agenda for global development, particularly when it comes to financing improved services and infrastructure for the poorest people in the world. Young people from Laos to Lagos aspire to become software developers, and smartphones are set to put unprecedented computing power into every corner of the earth. But the paradox is that many governments still have only rudimentary information technology infrastructure, and often can't find trained and skilled staff to design and run it.

As an example, in many African countries, the capacity

for central and regional government to work with digital tools is limited because it is common to find only a few people in the government department responsible for coordinating involvement and investment in, say, rural drinking water infrastructure and financing. Thus, they are easily stretched thin by the demands and the need to be experts on many aspects of IT and data systems. So, what is the solutions for these conditions?

Source: http://opensource.com/government/15/4/open-source-government-software
Submitted by: **Thomas Bjelkeman-Pettersson**

---

10 Years of Git: An Interview with Git Creator Linus Torvalds

Ten years ago this week, the Linux kernel community faced a daunting challenge: They could no longer use their revision control system BitKeeper, and no other Software Configuration Management (SCMs) met their needs for a distributed system. Linus Torvalds, the creator of Linux, took the challenge into his own hands and disappeared over the weekend to emerge the following week with Git. Today, Git is used for thousands of projects, and has ushered in a new level of social coding among programmers.

To celebrate this milestone, this article provides the behind-the-scenes story of Git, and tells us what Linus thinks of the project and its impact on software development. You'll find his comments in the story below. We'll follow this Q&A with a week of Git in which we profile a different project each day that is using the revision control system. Look for the stories behind KVM, Qt, Drupal, Puppet, and Wine, among others.

Source: https://www.linux.com/news/featured-

blogs/185-jennifer-cloer/821541-10-years-of-git-an-interview-with-git-creator-linus-torvalds
Submitted by: **Jennifer Cloer**

How to Install Linux on a Windows Machine With UEFI Secure Boot

When Windows 8 rolled up to the curb, Microsoft did its best to enforce a protocol known as Unified Extensible Firmware Interface (UEFI) Secure Boot. This was to be a modern replacement for the aging BIOS system, and would help ensure boot-time malware couldn't be injected into a system. For the most part, Linux has overcome those UEFI hurdles. However, with Windows 10, those hurdles could be returning.

This BIOS replacement, UEFI, caused some serious problems with "alternative" platforms. For some time, it was thought UEFI would render Linux uninstallable on any system certified for Windows 8 and up. So what are you to do when you have a new system and you want to install Linux? The answer isn't always simple.

Source: https://www.linux.com/learn/tutorials/821007-how-to-install-linux-on-a-windows-machine-with-uefi-secure-boot
Submitted by: **Jack Wallen**

Unity 8 Won't Be Very Different Visually from Unity 7

Unity 8 is not ready for regular use, but Canonical is improving it all the time. Users need to know that despite all the important changes and improvements announced for Unity 8, it won't be terribly different from the current implementation.

One of the things Canonical learned the hard way is the

fact that you don't radically change the desktop environment and hope that you made the right call. The company changed the desktop from GNOME 2 to Unity, for reasons that can be explained at a later date, but the change was too sudden. It didn't matter that users had some time to get used to Unity before the official launch. Most of them hated Canonical for that decision, and many still do.

A lot of the Ubuntu users are now used to Unity and the way it works. To be fair, the developers have greatly improved the desktop environment over the years, and it's now pretty much complete. The only thing that Canonical needs right now is to replace Unity 7 with an alien Unity 8, and some users in the community think that it's going to be a big change.

Then, how big it the change of Unity 8 from Unity 7?

Source: http://news.softpedia.com/news/Unity-8-Won-t-Be-Very-Visually-Very-Different-from-Unity-7-477711.shtml
Submitted by: **Silviu Stahie**

Linux Seeks Security, Unity

Linux is expanding its reach, promising to play a significant role in the Internet of Things. But the open source software needs more attention to interoperability, security, and its kernel – according to experts at the Embedded Linux Conference here.

Industry giants including Samsung and Panasonic are expanding new open source groups. Jaguar Land Rover came to the event saying it will make open source its connected car software, challenging competitors to do the same.

In IoT, Linux commands the gateway today, with ambitious efforts to pack it into end nodes. It already runs on microcontrollers, with some developers aiming at a MByte-size version of the open source operating system, one speaker said.

Intel is working on subsets of Tizen for IoT. “Today it’s still extremely challenging to run Linux on less than 8 MBytes of memory…[so in IoT] we don’t know where the compromises are yet,” said Dominique Le Foll, a Linux expert at Intel.

Source: http://www.eetimes.com/document.asp?doc_id=1326150
Submitted by: **Arnfried Walbrecht**

Growing, IoT Next Big Win

A new VDC Research study projects that Linux and Android will continue to increase embedded market share through 2017, while Windows and commercial real-time operating systems (RTOSes) will lose ground. The study suggests that the fast growth of IoT [Internet of Things] is accelerating the move toward open source Linux.

"Open source, freely, and/or publicly available" Linux will grow from 56.2 percent share of embedded unit shipments in 2012 to 64.7 percent in 2017, according to VDC's "The Global Market for IoT and Embedded Operating Systems." That represents a CAGR of 16.7 percent for open Linux, says VDC.

As that indicates, it also looks at Android's share of this market. Interestingly, Android does not compete much with vanilla Linux, but rather with the embedded version of Windows.

In the general embedded market, Android is eating into

the share of Windows Embedded more than that of Linux, says VDC. Android is moving quickly into areas where Windows has generally outperformed Linux, including automotive infotainment, medical devices, military handhelds, and, to a lesser extent, retail and signage.

Source: http://www.computerworlduk.com/blogs/open-enterprise/embedded-3605044/
Submitted by: **Arnfried Walbrecht**

Google makes deploying software on its cloud a trivial task

According to a Google blog post, Cloud Launcher offers you “a continuum of compute options -- from high performance VMs and container-based services to managed PaaS -- so you can choose the most suitable option."

With the Google Cloud Launcher, you can launch more than 120 popular open-source packages. These programs have been configured by Bitnami, a company that provides a one-click install library of popular server applications and development environments, and Google Click to Deploy – Google's own set of easy-to-setup programs.

The Cloud Launcher makes it simple to find which program you want to set up as well. You can either search for them, or look at the offerings by category.

Google claims that "deployment is incredibly straightforward: users simply select a package from the library, specify a few parameters, and the package is up and running in a few clicks." Indeed, a quick check found that it was easy to set up applications. In addition, Cloud Launcher tells you, before you start running, what

processors you'll get, how much disk space your app will take up, and how much you can expect to pay for your new Google Cloud Platform every month.

Source: http://www.zdnet.com/article/google-makes-deploying-software-on-google-cloud-trivial/
Submitted by: **Arnfried Walbrecht**

Linux needs interoperability, security: Experts

More and more large organizations are becoming interested in Linux, but experts at the recent Embedded Linux Conference say the operating system needs more work in the areas of interoperability and security.

For example, Intel Inc. is working on a subset of Tizen (an operating system based on the Linux kernel) and the GNU C Library implementing the Linux API. Tizen targets a wide variety of devices including tablets, smartphones, PCs, and infotainment devices. Intel is working on Tizen for its Internet-of-Things initiative.

Linux is seen as a "gateway" to IoT, but Dominique Le Foll, a Linux expert for Intel, told electronics industry news site eetimes.com, Intel is having difficulties with Linux.

He said it is still "extremely challenging to run Linux on less than 8Mbytes of memory… we don't know where the compromises are yet."

Source: http://www.itworldcanada.com/post/linux-needs-interoperability-security-experts
Submitted by: **Arnfried Walbrecht**

Tiny, stackable, Linux-based IoT module hits Kickstarter

On Kickstarter, Onion launched a tiny, Linux-based "Omega" IoT module, along with a dock, stackable expansion modules, a cloud service, and web app tools.

Onion's Omega joins a growing number of single-board computers and computer-on-modules for Internet of Things applications that have tapped Qualcomm's MIPS-based, WiFi-enabled Atheros AR9331 system-on-chip. For a pledge of $25, Onion's Kickstarter campaign offers the Omega computer-on-module combined with a "dock" that turns it into an sandwich-style single-board computer.

The Omega's standard dock includes a USB port, power supply, LED, push buttons, and a 30-pin expansion bus that accommodates self-stacking Omega expansion modules. A $35 Kickstarter package lets you add one of the Omega's stackable expansion modules (described further below), or an Arduino Shield dock that enables you to "use your existing Arduino shields with the Omega," says Onion.

These packages, as well as other kits ranging up to $199, are similarly expected to ship in August. More advanced robot and drone kits are due in Oct. 2015 and Jan. 2016, respectively, says the project. Other open source kits shown on the site include a spy cam, LED matrix art array, ping pong blaster, and a tweet printer.

Source: http://linuxgizmos.com/tiny-stackable-linux-based-iot-module-hits-kickstarter/
Submitted by: **Arnfried Walbrecht**

Total boosts Linux Pangea supercomputer with 4.4 petaflops of compute power

Oil and gas giant Total has chosen SGI to upgrade its supercomputer, adding 4.4 petaflops of compute power to

assist in exploration and production of resources.

The company launched the high performance computing (HPC) platform in 2013, dubbed Pangea, which runs on Linux Enterprise Server. Built on SGI's ICE X technology, it was claimed that the 2.3 petaflop supercomputer was one of the most powerful in the world, housing over 110,000 cores, using Intel Xeon E5-2600 processors.

The upgrade announced today will add a further 4.4 petaflops of compute due to the latest ICE X system, which uses the most recent version of Intel's E5-2600 chips, launched in September. It will also feature another 9.2 petabytes of storage, with a total of 589 terabytes of memory built across 8 M-Cells.

The HPC, located at Total's Jean Feger Scientific and Technical Centre in Pau, France, will use closed-loop airflow and warm-water cooling to lower overall cooling requirements. Once fully upgraded, the system will use 4.5 megawatts of power.

Source: http://www.computerworlduk.com/news/it-business/3605998/total-boosts-linux-pangea-supercomputer-with-44-petaflops-of-compute-power/
Submitted by: **Arnfried Walbrecht**

System76 unveils all-new Broadwell-powered Lemur -- an affordable Ubuntu Linux laptop

While many computer manufacturers are in a race to the bottom -- both in price and quality -- some makers continue to produce reliable high-quality machines. One of these manufacturers is System76. If you aren't familiar, it manufactures and sells desktops and laptops running the Ubuntu operating system. In other words, Linux fans can buy one of these machines and have it running the

Linux distro out of the box -- no need to format the drive to remove Windows.

Today, the company announces the all-new Lemur -- a laptop with a Broadwell processor. The 14.1 inch laptop not only has solid specs, but an affordable price tag too. If you are a Linux fan in the market for a new laptop, you need to take a look.

"The Lemur is a new laptop class for System76. It's designed to achieve long battery life and affordability, while maintaining the high quality standards our customers have come to expect", says Carl Richell, CEO and Founder, System76.

Speaking of long battery life, the company claims it can easily last an entire flight across the continental United States -- California to New York. While this is impressive, your mileage may vary depending on the type of computing you do.

Source: http://betanews.com/2015/04/02/system76-unveils-all-new-broadwell-powered-lemur-an-affordable-ubuntu-linux-laptop/?utm_source=feedburner&utm_medium=feed&utm_campaign=Fee+-+bn+-+Betanews+Full+Content+Feed+-+BN
Submitted by: **Arnfried Walbrecht**

Linux Foundation's new mission: Cheap, easy Web encryption

After ramping up efforts to fund auditing and full-time development for crucial Internet protocols, the Linux Foundation now throws its weight behind an effort to make encryption on the Web inexpensive and easy to deploy.

The Let's Encrypt project, originally created by the ISRG

(Internet Security Research Group), aims to provide free, validated TLS certificates to anyone who owns a domain, and by providing tools to make deploying and updated certificates as simple as a single command-line operation.

Deploying TLS encryption certificates can be both complex and costly, but the complexity may be the bigger stumbling block. Even Microsoft and Google have had trouble keeping their certificates up to date. Azure suffered an outage, and Gmail hiccuped because of expired certificates that weren't renewed.

With Let's Encrypt, said Linux Foundation executive director Jim Zemlin, the idea is to make installing and upgrading certificates "as simple as apt-get," referring to the semi-automatic command-line software updater featured in Linux.

Source: http://www.infoworld.com/article/2908155/encryption/linux-foundations-new-mission-cheap-easy-web-encryption.html
Submitted by: **Arnfried Walbrecht**

IBM Tests Mobile Computing Pioneer's Controversial Brain Algorithms

For more than a decade, Jeff Hawkins, founder of mobile computing company Palm, has dedicated his time and fortune to a theory meant to explain the workings of the human brain, and provide a blueprint for a powerful new kind of artificial intelligence software. But Hawkins's company, Numenta, has made little impact on the tech industry, even as machine learning has become central to companies such as Google.

Now, one tech giant is finally taking an interest.

IBM has established a research group to work on

Numenta's learning algorithms at its Almaden research lab in San Jose, California. The algorithms are being tested for tasks including interpreting satellite imagery, and the group is working on designs for computers that would implement Hawkins's ideas in hardware. Hawkins says that around 100 people are working on the project, known internally as the Cortical Learning Center.

Source: http://www.technologyreview.com/news/536326/ibm-tests-mobile-computing-pioneers-controversial-brain-algorithms/
Submitted by: **Arnfried Walbrecht**

---

elementary OS team releases 'Freya,' a Linux distro you'll want to see

This weekend, the elementary OS team released the latest version of their Linux distribution, focused on usability and high-quality visual design, and code named "Freya." And, if I were a member of the Ubuntu or Linux Mint teams, I would take serious notice.

I've been running this new version of elementary for a few days (a review will be coming after I've used it for a full week, so I can really get a chance to see what it's like to live in it). But I must say… this is one of the most polished releases of any Linux distributions I have ever had the pleasure to use.

That's not hyperbole. I have no reason to exaggerate the quality of this release – I'm an openSUSE and Android guy, through and through. But "Freya" is damned impressive.

This is the first release of elementary OS since the team released their last version ("Luna") back in the summer of 2013.

"With Luna, much of our work was building the first version of our desktop environment Pantheon," Cassidy James Blaede, UX designer at elementary, said in an elementary blog post. "With Freya, we were able to dial in on the details like super-subtly rounded corners on the display, and the new animations for maximizing. Both are small, yet make the whole experience more polished. We were also able to focus much more effort on the included apps, delivering new features for existing apps and new apps alike."

Source: http://www.networkworld.com/article/2908970/opensource-subnet/elementary-os-team-releases-freya-a-linux-distro-youll-want-to-see.html
Submitted by: **Arnfried Walbrecht**

Raspberry Pi gains an E-paper display

A Kickstarter project is pitching a HAT add-on for the Raspberry Pi that provides a 2.7-inch E-paper display, as well as a battery-backed real-time clock.

For educators, one of the coolest things about the Raspberry Pi is the HDMI port, which let you easily plug in to a monitor. But for embedded gizmos, a more modest display is often more suitable. It doesn't get much more modest than Percheron Electronics' E-Paper HAT Display, a Raspberry Pi add-on board that drives a 2.7-inch, 264 x 176-pixel E-paper display from Pervasive Displays.

Percheron is just shy of its $15,241 Kickstarter goal and has already sold out two early-bird rounds. The E-Paper HAT Display is currently available for a low of 39 UK Pounds (about $57) or 69 Pounds ($101) for a two-pack. Shipments are due in September.

Source: http://linuxgizmos.com/raspberry-pi-gains-an-e-

paper-display/ Submitted by: **Arnfried Walbrecht**

Is DARPA's new search engine, Memex a Google-killer?

When we look at the history of computing, it features a string of organizations that, for some time, looked as if they were deeply entrenched in our lives and we would never be able to do without them.

For example, IBM and Microsoft looked liked that. In recent times, it has been Google and Facebook.

Sometimes they look indisputable because of the the narrow territory they occupy. When they fall, it is because the situation has changed drastically – and not because someone has captured their territory.

For several years, Linux enthusiasts proclaimed "this will be the year that Linux finally competes with Windows on the desktop!"; however, every year it did not happen.

Eventually Linux smoked Microsoft under the brand name Android, when 'Desktop' gave way to 'Mobile'.

Google has been the heavyweight and king of web search since the late 1990s. All efforts to throw Google out of the market have failed. Not only does it have a strong hold on the market share but it has also been able to keep all challengers at bay – ranging from awkward tech colossus to smart and disturbing startups.

Google will not surrender its territory to a Google duplicate, but may, one day, find that its territory is not the same as it was previously.

The web is getting broadened and darker, and Google, Bing and Yahoo are not able to search most of it.

They don’t search the sites that have been asked to be ignored, or that cannot be found by following links from other websites (the vast, virtual wasteland known as the Deep Web). They even don’t search the sites on anonymous, encrypted networks like Tor and I2P (the so-called Dark Web).

Source: http://www.techworm.net/2015/04/is-darpas-new-search-engine-memex-a-google-killer.html
Submitted by: **Arnfried Walbrecht**

Open source patent protection extended to popular software

OIN (the Open Invention Network) has announced important changes to its definition of what constitutes “the Linux system."

Licensing community participants were advised yesterday that, as of June 25, the Linux System Definition will include more than 100 new packages and updated versions of nearly 70 existing ones. This refresh brings Docker, Puppet, Nginx, the Ceph storage system, and the full LibreOffice suite, under the OIN umbrella, as well as the Debian packaging tool APT, the Java build manager Maven, the reference implementations of the Go and Lua languages, and a number of other development tools and libraries. Coverage for Red Hat’s OpenShift project will also be significantly enhanced.

Source: http://www.infoworld.com/article/2910934/open-source-software/open-source-patent-protection-extended-to-popular-software.html
Submitted by: **Arnfried Walbrecht**

Command & Conquer

Storage & Encryption

Lucas Westermann

As many of you may have gathered by now, I work in IT. As my work often requires me to visit clients and have solutions or demos ready to run on their computers, I always have a USB thumb drive with me that contains some of my tools and demos. The real trick to this is having a thumb drive that's readable on any computer with minimal fuss and without file-size restrictions, as well as keeping sensitive documents for other clients safe. This month, I'll cover exactly that.

Cross-Platform Storage Format

The de facto standard for hard drive formatting is FAT32. This was developed by Microsoft back in 1996. It includes a file-size limit of 4GB - this means that no individual file can be larger than 4GB. The restriction is increased in FAT32+, though support for that standard can be hit-and-miss. In the case of backups or archives, this can be a problem. NTFS (also by Microsoft) can be read by all major OSes, but the newest versions of Mac OS X lack write support by default. HFS+ can be read by Linux, but not by Windows (without paid software at least). This is where exFAT comes in.

ExFAT stands for "extended File Allocation Table" (extended FAT), and is the default formatting for SDXC cards larger than 32GB. It was originally introduced in 2006, and is supported in all modern OSes (this excludes "legacy" OSes like Windows XP). Windows and Mac both support this out of the box. Some Linux distributions, however, do not have the utilities installed out of the box

(they will be available in the repositories though). In Ubuntu, install the exfat-fuse and exfat-utils packages from the universe repository.

Formatting the drive is the exact process used to format any drive, just be sure to select exFAT.

Encryption

VeraCrypt

For a long time, I relied upon TrueCrypt for creating encrypted file containers. Unfortunately, development for TC has stopped, and, for a time, no similar software appeared. I recently stumbled upon a fork of TC called VeraCrypt that seeks to solve the security issues of TrueCrypt (and according to the audit of TC, seems to have solved the largest issue).

VeraCrypt offers two options - encrypted file containers, or encrypted partitions/devices. File containers are simply encrypted files that can be mounted through VeraCrypt, and act as local USB drives. No unencrypted data is written to disk, but it is stored in RAM (on-the-fly encryption/decryption). Encrypted partitions are, as one might expect, a completely encrypted partition. If you plan on storing unencrypted files alongside encrypted ones, I would recommend the file containers. If you want a fully encrypted device, choose the partition. Keep in mind that the partition can be mounted only through VeraCrypt. You'll either need a portable install with you, or a system with it already installed.

My personal preference is to have multiple file containers (one per client), and a portable copy of VeraCrypt installed on my USB drive. You need to have a portable copy of VeraCrypt for each OS - this is typically done by copying the extracted files during installation. That way, I can run the application off the USB drive, and have access

to all the file containers I need.

GPG

Gnu Privacy Guard (GPG) is an encryption standard that can be used for file encryption (as well as the encryption of messages, such as email). It uses public key cryptography (existence of both a private and a public key to encrypt). Without going into extreme amounts of detail, a file or message is essentially encrypted using a public key (the key you can give away and share). And decryption requires the use of the matching private key. If someone were to send me an encrypted message, it must be encrypted using my public key, but I would need to decrypt it with my private key.

This encryption is typically useful for messages or securing a single file or group of files. It is not on-the-fly encryption and decryption, and decrypting a file will result in the unencrypted data being saved on your hard drive. It also does not encrypt folders. However, if you're sharing sensitive data via Email, it may be better to use GPG than VeraCrypt or an encrypted ZIP Archive.

For instructions on using GPG encryption, see the further reading section.

Hopefully this article will be helpful to some - especially those who have to switch between various OSes like I do. If you have any questions, issues, or requests, please let me know at lswest34+fcm@gmail.com. This includes requests for articles.

Further Reading

- https://veracrypt.codeplex.com/wikipage?title=Portable%20Mode - How-To on portable VeraCrypt
- https://veracrypt.codeplex.com/discussions/572934 - How-To on cross-platform portable VeraCrypt.

- https://www.gnupg.org/ - GPG homepage

LibreOffice, Part 48: Insert and Link Sheets

Elmer Perry

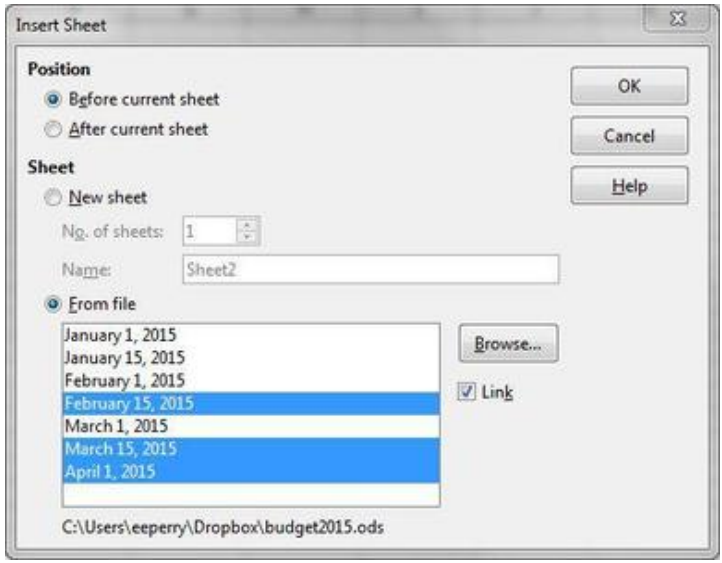
Linking to data on the web is often useful, but, sometimes you need to just get data from a sheet in another Calc file. You can just import or link the whole sheet from another file. You do this through the Insert Sheet dialog. Another choice, with a little more power, is to create named ranges of the data you need to link to your new sheet. You can do this through the Navigator as we did with the web data, but there is another method for inserting through the External Data dialog.

Difference Between Insert and Link

Since our main topic concerns linking data from other sheets, you need to know the difference between inserting a sheet and linking a sheet. When a sheet is inserted, you can change the cells in both sheets and changes in one will not affect the other. However, when you create a link in sheet 2 to sheet 1, changes to cells in sheet 1 will override changes in sheet 2 when sheet 2 is updated. Changes in sheet 2 have no effect on sheet 1.

Link/ Insert Whole Sheet

Sometimes, you just need the entire sheet from a spreadsheet file in your new spreadsheet file. If you want to make changes to the sheet, and have those changes stay, you will need to import the sheet. However, if you need any changes to the original sheet to apply to the new sheet also, you need to create a link. From the menus, Insert > Sheet will bring up the Insert Sheet dialog.



You have two choices for the position of the new sheet within the spreadsheet file. You can place it before or after the currently selected sheet. The Position options allows you to select where you want the imported / linked / new sheet(s) located.

You have the choice of creating new blank sheet(s) or select an existing sheet from another file. The settings under “New Sheet” allows you to create one or more new blank sheets. The settings under “From sheet” let you import, and optionally link, a sheet from another file.

For the “New sheet” options, you can specify a number of sheets to add and a name for the sheet (if you’re adding one sheet). If you are creating more than one sheet, the option for the name is grayed out. The sheets are created with the default sheet prefix and a numeric iteration. You can set the default sheet prefix in Tools > Options > LibreOffice Calc > Defaults.

For the “From file” options, you use the Browse button to locate the file containing the sheet(s) you want to import

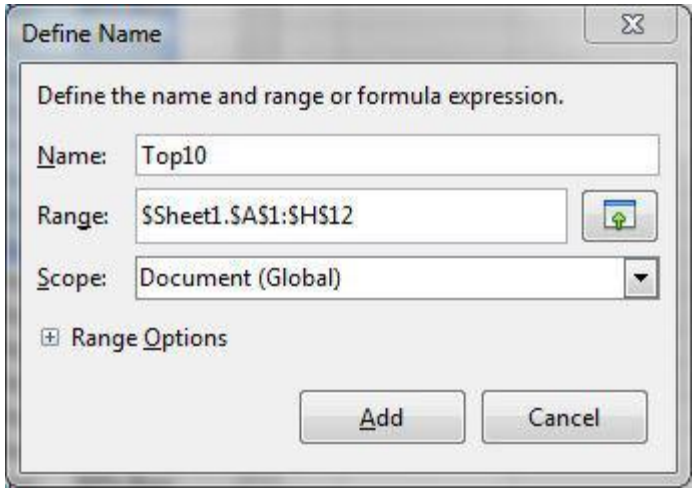
into your new spreadsheet file. Once you open the file containing the sheets you want to use, a list of sheets will appear in the list box. Select the sheet(s) you want in your new file. To select sequential sheets, click on the first one, then hold SHIFT while clicking on the last one. To select non-sequential sheets, click on the first one, then hold CTRL while clicking on the others. Once you have your sheets selected, you can check the Link check box to link to the sheet(s) rather than just import a copy of them. Remember, when linked, the data will update from the original sheet.

Click the OK button to import / create the sheets using the options you have selected.

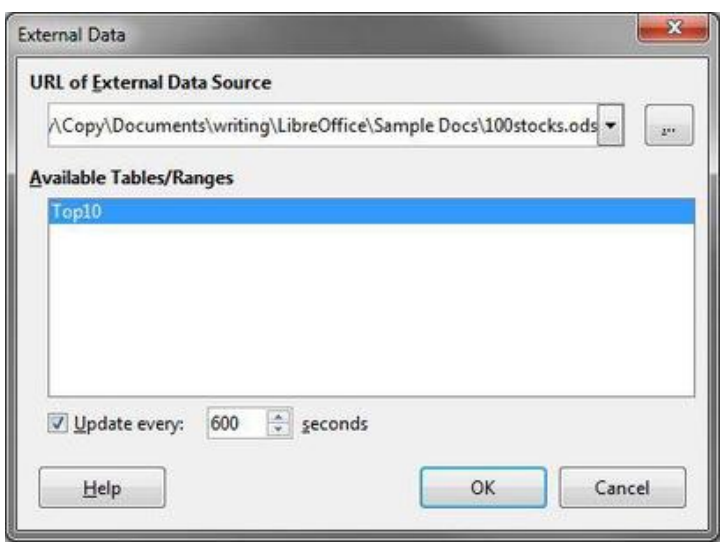
Link Part of a Sheet

When I showed you how to link to data on the web, I used the Navigator to link the data. Today, I am going to show you another method for linking data, but this time we will link data from another file rather than from the web.

When linking to just a section of a sheet, you need a way to define the section of the sheet you want to link. This is done by creating a named range. We discussed named ranges back in part 42 (Full Circle Magazine, issue 90). We will create a named range in order to link to that portion of the sheet.



For example, let's say you want to link to just the top ten stocks from the top 100 stocks we pulled from the web last time. Open up the 100 stocks file linked to the data on the web. Select the range A1:H12. Insert > Name > Define. The Define Name dialog will show. Give the range a name like “Top10”. Remember that the name cannot contain spaces. The range should fill in with the range you selected. Leave the scope as “Document (Global).” Click the Add button, and save the file.



Create a new spreadsheet file, File > New > Spreadsheet. We will use the External Data dialog instead of the Navigator this time. Insert > Link to External Data opens the dialog. The drop-down list shows a list of recent documents. Select the top 100 stocks file. If the document you need does not appear in the drop-down list, you can click on the ellipse (…) button to browse to and select the file. Once you select your file, the named ranges appear in the list box. Select the named range Top10 from the stocks file to link to the top 10 stocks. You can set how often the file should update when open by setting the "Update every" field. Keep in mind the number is in seconds.

Click OK and you now have a link to the top 10 stocks. The data will update from the original sheet every time you open the file or at the specified update interval.

You can link to a web page using the External Data dialog as well. Just type in or paste the URL into the drop down box and press enter. You will get the import data dialog discussed in the last article. You will get a list of link names. Unlike the Navigator method, there is no way to preview which element is the one you need. It becomes a matter of trial and error (mostly error). This is why I recommended the Navigator method as the best for linking to web data.

You can also use the Navigator to link to named ranges in other documents. Just drag the ranged range into your new document.

Not only can we link to data on the web, but we can also link to data in other Calc files. We can import or link entire sheets using the Insert Sheet dialog. If we need just a portion of a sheet, we can create a named range in the original sheet. In a new sheet, we can link to the named range in the original sheet through the External Data

dialog. Always keep in mind the difference between linking and importing. Imported data is not affected by changes in the original, but linked data is affected by changes in the original.

FCM 96 - How To - Program in Python - Part 58

By G.D. Walters

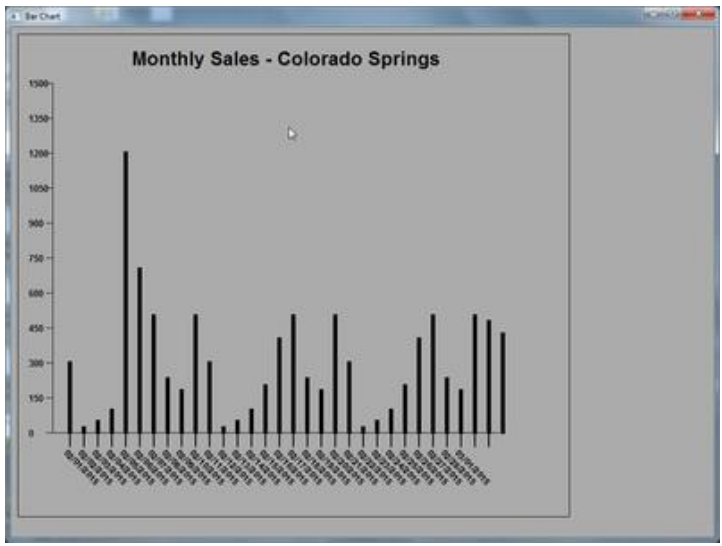
Creating our own charting library…

Last time, we discussed reading and using data from an Excel file directly. If you remember, my boss (from my "day" job) had a massive spreadsheet that if one calculation failed, it caused the entire process to abort. Well, I created a database from that spreadsheet that was easy to get a report from. However, the original spreadsheet created pretty charts and graphs that his bosses liked to see. So I undertook the task to create charts so everyone would be happy.

After spending about two days digging into the existing charting/graphing packages already available for Python, most free, most of them output directly to a file, like a pdf file or some sort of a graphics (jpg, png, svg) file. What I was looking for was one that would go directly to a wxPython frame or panel so it can be displayed inside a GUI program. I found one solution, but it required so many interdependent libraries that the possibility of just giving the application on a flash drive quickly became nil.

So, being the pig-headed, tenacious, never-say-die kinda guy that I am, I decided to write one on my own. The original goal was that it was to do (at least) bar-charts and maybe in the future line charts and/or other types. It also should eventually be able to do colours, but just plain black bars would suffice for the time being. It should be standalone in general so that it could be called as a library. It wasn't supposed to be so generic that it gets complicated, just dates along the horizontal (bottom) axis,

values along the vertical axis and bars that represent the daily sales for that period. In order to keep the chart somewhat neat, the dates should be angled so that they don't overwrite each other. So, what I came up with will be presented here. Below, is a sample output of the code:



Again, not fancy, not terribly pretty, but it does the job. If it needs to be prettier later on, then I can work on it down the road. The first thing I had to do was pull out my wxPython documentation to remember the graphic commands. In able to draw graphics, we use the a "dc" or Device Context. It's sort of like a blank canvas that we can draw points, lines, and text to. wxPython offers 9 different types of dc objects and I chose the wx.PaintDC which works from the OnPaintEvent. We will use some very basic commands to do our drawing and painting. These are:

- dc.DrawLine
- dc.SetPen
- dc.SetFont
- dc.DrawText
- dc.DrawRectangle

- dc.DrawRotatedText
- dc.GetFullTextExtent

Those are the only wxPython routines we will use, though there are many others that would make our program much prettier. We will combine these commands into our own "logical" routines like, DrawBars, DrawAxis, DrawValues, and so on. While I could have done it in one or two large routines, I wanted to break them out into routines that make sense for the teaching moment. So let's get started looking into the code. Create a file called mygraph.py. I couldn't come up with anything pithy, since PyChart, PyGraph and the like are all taken. Maybe if I had a bit more time, I'd come up with something, but that's not important. Let's get started. First, we'll do the imports as we always do.

```
#!/usr/bin/python
# mygraph.py
import wx
from datetime import date, datetime, time
import time
import math
```

Obviously, we need to import he wxPython library and the math library will help us with some of the calculations. The datetime and time libraries are used to do the date calculations for the horizontal axis labels.

Something to keep in mind as we go from here…When you think about drawing on a context, the upper left corner of the container window (our dc) is X = 0, Y = 0. X is the horizontal axis and Y is the vertical axis. The closer we get the lower right corner, both numbers go higher. In our program, we will actually start by drawing a box that defines our charting area which starts at upper left X = 10, Y = 10 and end with lower right at X = 800, Y = 700. However, before we get to that part, we have to define a

class to handle the routines and the **init** routine. Hopefully you remember these from earlier sessions.

Here is the class definition and the **init** routine:

```
class Line(wx.Frame):
   def _init_(self, parent, id, FrameTitle, IncomingData, ChartTitle):
      wx.Frame._init_(self, parent, id, FrameTitle, size=(1024, 768))
      self.Bind(wx.EVT_PAINT, self.OnPaint)
      self.BoxWidth = 790
      self.BoxHeight = 690
      self.ChartTitle = ChartTitle
      self.data = []
      self.SetData(IncomingData)
      self.Centre()
      self.Show(True)
```

Our class is called Line and we will be creating a wxFrame to do our drawing. This could also be a panel within a frame or any number of other options. My choice was to have a Frame pop up with our chart data on it. When the class is first instantiated, the **init** routine is called with the name of the parent object, the id of that object, the title of the frame (in the title bar), the data that we want to chart and finally the title of the chart itself. Next we create the wx.Frame object that is 1024x768 pixels in size. Next we bind the paint event (which is called everything the frame is created, moved, covered, uncovered, etc.) to our event routine OnPaint. Remember, since this is inside of a class we use the “self.” to say the routine belongs to the class not somewhere else. We set some variables (BoxWidth, BoxHeight, ChartTitle, data) for use later. After we set self.data to an empty list, we call a routine called SetData to find our data scale, which we will discuss further down. Finally, we set the frame to be centered in the screen and call the Show routine. This will automatically call the OnPaint routine since we are creating the Frame.

Next we will write a routine that will create a box that shows the area that we want to constrict our graph to. This is not a clipping or constraining box, it is simply to draw the eye to what we want the user to look at.

```
def DrawBox(self,dc):
    #Horizontal
    dc.DrawLine(10,10,800,10)
    dc.DrawLine(10,700,800,700)
    #Vertical
    dc.DrawLine(10,10,10,700)
    dc.DrawLine(800,10,800,700)
```

This is fairly simple. We pass the dc of the frame, then draw four lines. The DrawLine function parameters are:

```
dc.DrawLine(StartX, StartY, EndX, EndY)
```

Not really difficult. We will be using the DrawLine function several times throughout the program. Next we will create a routine that will draw the X (horizontal) and Y (vertical) axis lines on the screen. We again pass the dc of the frame into the routine.

```
def DrawAxis(self,dc):
    #Horizontal
    dc.DrawLine(60,580,700,580)
    #Vertical
    dc.DrawLine(60,580,60,80)
```

Since we just discussed the DrawLine method, there's nothing very out of the ordinary here. We are drawing a line 580 pixels down the Frame that starts at X = 60 and ends at X = 700. Then we draw a line that starts at X = 60 Y = 580 and goes up to X = 60 Y = 80. This one is drawn from the bottom up, but we could have drawn it from the top down.

Next we will deal with the DrawTitle routine. Once again, we pass the dc of the frame as well as the text we want to draw. During this process, think of drawing text rather than printing text. It's a very minor thing, but it will help.

```
def DrawTitle(self,dc,txt):
    dc.SetFont(wx.Font(20,wx.DEFAULT,wx.NORMAL,wx.BOLD))
    dc.SetPen(wx.Pen(wx.NamedColour('black'),20))
    #Get the length of the text to draw
    vals = dc.GetFullTextExtent(txt)
    # Returned (Width,height,Decent,externalLeading)
    #Get the left position (x) to draw centered text
    txtleft = (self.BoxWidth-vals[0])/2
    dc.DrawText(txt,txtleft,30)
    # Reset the pen size and colour
    dc.SetPen(wx.Pen(wx.NamedColour('black'),2))
```

This routine is longer than most of the others, but part of that is the comments I put in. The first two lines set the font and the pen style that we will be using. In the first line (SetFont), we define the font to be the "default" font, 20 points, not italic and bold. Next we set the colour of the pen to black and the width to be 20. Now we need to figure out the width of the text that we will be drawing so we know how to center it in the box. We get this information by calling the GetFullTextExtent with the text that we will be drawing using the font, font size, pen width and so on that we just defined. The tuple that is returned contains Width, Height, Decent (how far down letters like "g" and "y" will go below the base line) and any leading space. For our purposes, all we are concerned with the width. If you remember, we defined the width of the box back in the **init** function as 790. To find the center of our text within our box we take the box width minus the width of the text and then divide it by 2. That will be the X value we use to draw our text. Finally, we reset the pen size and colour. Rather than use some default values we pick out of nowhere, we could have called the dc.GetPen function before we started, but when

I started the project, I didn't think about it.

Our next routine will draw the tic lines along the horizontal axis at the bottom of the chart. We want them to be equidistant along the line. We pass (as usual) the dc and a value I called dcount which is the number of dates we want to show. Since the number of days in any given month can range from 28 to 31, I wanted to be a bit dynamic. We simply use a for loop to count the number of lines to draw, which one to draw and where. If you have been carefully paying attention, we will start the lines at position 85 and it will be 20 pixels high and they will be 20 pixels apart.

```
def DrawDateTicks(self,dc,dcount):
    for cntr in range(1,dcount+1):
        dc.DrawLine(65+(cntr*20),580,65+(cntr*20),600)
```

When we get around to drawing the dates into the chart, we want to draw the text on an angle. That way, the text doesn't draw over itself and, well let's admit it, looks cool. For this we will use the DrawRotatedText function. The function takes the text that we want to have drawn, the X and Y location as a starting point and the angle we want the text to be drawn. In this case, we want the text to be rotated anti-clockwise by 45 degrees which we enter as "-45". We will set the font and pen parameters each time the text is drawn. We'll deal with the actual draw date function in a little bit.

```
def DrawRotText(self,dc,txt,x,y):
    dc.SetFont(wx.Font(10,wx.DEFAULT,wx.NORMAL,wx.BOLD))
    dc.SetPen(wx.Pen(wx.NamedColour('black'),20))
    dc.DrawRotatedText(txt,x,y,-45)
```

We will also want to draw the values along the vertical axis showing tics along the way. If we had the same range of data each time, it would be very easy to do. However,

reality shows that the data range of our chart could vary from run to run. One time, the highest value could be 300. The next time it could be 3000. How could we create a generic routine that would account for this? I will try to explain my mindset here.

You might have wondered why I chose the value of 500 for the vertical axis when we drew the line from 80 to 580 (or actually 580 to 80). I chose to use a 500 pixel "view port" to contain our values. That way, we can create a scaling value based on an offset of 500.
Let's say that for a given run that our highest value is 395. We can simply draw a bar that is 395 pixels high to represent that value. The next run, our highest value is 2,345. If we try to draw the bar to its full height, it would disappear off the top of the chart. In order to show the value, we can round the value to the nearest 500, which would be 2500 and then set that as the top value of our axis. We then can scale the value by dividing 2500 by 500 which gives us a "scaling factor" of 4. Now if we take our data values and divide each one by our scaling factor, we can then plot the values that they will fit within our graph.

So, we need to find the highest value within our data and round that up to the nearest 500. So 375 would be 500, 3750 would be 5000 and so on.

```
#=========================================
# Round up to the nearest 500
#=========================================
def roundup(self,x):
    return int(math.ceil(x/500.0))*500
```

Next, we need to decide what kind of data we are going to use. You will see further down the program that I provide two different types of data in lists. One assumes that the date range we will use, along the X axis, is data for

October, but you can easily follow that code (shown in a few moments) and change it to whatever month you wish. The second data list, is more generic and provides both a date and a value as a list of tuples. This allows for data to be passed for any time period. The date is a string and the value is either an integer or a float. The SetData function will look at the first value within the data list and to determine if it is a tuple. If it is, we assume that the data structure of the list is the second option, if not, it is the first.

If it is a tuple, we create two lists, one for the dates and one for the values. We then walk the list splitting the data between the two lists. Once we have that done, we then find the highest value (max(self.ValList)) and send it the roundup function (shown above) so we can get our scaling value. If the data isn't in tuples, then we clear BOTH lists and do the same steps as above.

```
def SetData(self,DataToUse):
    if type(DataToUse[1]) is tuple:
        self.DateList = []
        self.ValList = []
        for l in DataToUse:
            self.DateList.append(l[0])
            self.ValList.append(l[1])
        self.HiValue = self.roundup(max(self.ValList))
        self.ScaleValue = self.HiValue/500
    else:
        self.ValList = []
        self.DateList = []
        for l in DataToUse:
            self.ValList.append(l)
        self.HiValue = self.roundup(max(self.ValList))
        self.ScaleValue = self.HiValue/500
```

Now that we have our scale value we can draw our tics and the values that will represent our vertical axis. We again use a for loop, this time from 580 to 30 with a step of -50 to work our way up the line and draw a 10 pixel

line. Next we set the font (just in case it gets changed somehow) and draw the value of each of our values.

```
def DrawValues(self,dc):
    c2 = 0
    for cntr in range(580,30,-50):
        dc.SetPen(wx.Pen(wx.NamedColour('black'),1))
        dc.DrawLine(60,cntr,50,cntr)
        dc.SetFont(wx.Font(10,wx.DEFAULT,wx.NORMAL,wx.BOLD))
        dc.SetPen(wx.Pen(wx.NamedColour('black'),20))
        dc.DrawText(str(c2),26,cntr-7)
        c2 = c2 + (50 * self.ScaleValue)
```

The next function we will look at is the DrawBars function which will draw the actual bars that represent the data for each of our values.

```
def DrawBars(self,dc):
    dc.SetPen(wx.Pen(wx.NamedColour('black'),5))
    for cntr in range(0,len(self.ValList)):
        dc.DrawRectangle(84 + (cntr* 20),580,2,self.ValList[cntr]/-self.ScaleVa
```

Now we get into the routines that will create the date tics along the X axis if we choose to have a simple list of data without including the dates. We have two support routines, one called DateToStamp and the other Timestamp2Date (Yes, I got lazy when I wrote this one). Rather than going through a bunch of complicated DateTime routines to determine the number of days in any given month, I'm going to use a start date and an end date, convert both of those to Unix timestamps to get the proper day of month within the sequence. I've shown you the DateToStamp routine before and the Timestamp2Date simply reverses the process.

```
#=========================================
#  Convert mm/dd/yy to unix timestamp
#=========================================
```

```
    def DateToStamp(self,x):
        x = x+" 00:00:00"
        return(time.mktime(time.strptime(x, "%m/%d/%Y %H:%M:%S")))
    #=================================================
    #  Convert mm/dd/yy to unix timestamp
    #=================================================
    def Timestamp2Date(self,tstmp):
        return datetime.fromtimestamp(int(tstmp)).strftime('%m/%d/%Y')
```

The next routine takes the start date and end date, as we discussed a moment ago, converts them to Unix timestamps, then adds 86400 (the number of ticks in a 24 hour period) to make sure we get the last date in the sequence, then uses another for loop to draw the rotated text where we want it.

```
    #=================================================
    #  Draw the dates in rotated text
    #=================================================
    def DrawDates(self,dc,startdate,enddate):
        sd = int(self.DateToStamp(startdate))
        ed = int(self.DateToStamp(enddate))
        ed = ed + 86400
        stp = 1
        for cntr in range(sd,ed,86400):
            dt = self.Timestamp2Date(cntr)
            self.DrawRotText(dc,dt,65+(stp*20),600)
            stp = stp + 1
```

We are now at the OnPaint event handler that calls all the helper routines we dealt with so far. Remember, by using the PaintDC, every time the frame is moved, re-sized, covered or uncovered, the OnPaint event handler is called, thereby assuring our graph will be persistent.

First, we get an instance of our dc, and then we call the DrawBox, DrawAxis, DrawTitle, and the DrawDateTicks routines. We then determine if the DateList list (created in the SetData routine called from **init** routine) is empty or if it has dates for us to draw. If so, we call the DrawDates

routine with the proper values. We then call the DrawValues routine and finally the DrawBars routine. Now you should understand why I broke everything down into little bitty chunks.

```
    #=====================================================
    #  Main routine
    #=====================================================
    def OnPaint(self,event):
        dc = wx.PaintDC(self)
        self.DrawBox(dc)
        self.DrawAxis(dc)
        self.DrawTitle(dc,self.ChartTitle)
        ## Date Tics and dates
        self.DrawDateTicks(dc,31)
        leng = len(self.DateList)
        if leng > 0:
            sd = self.DateList[0]
            ed = self.DateList[4]
            self.DrawDates(dc,sd,ed)
        else:
            self.DrawDates(dc,"02/01/2015","03/01/2015")
        # Value Tics - Draw 10 tics
        self.DrawValues(dc)
        # Finally Draw the data bars
        self.DrawBars(dc)
```

The last thing we have to look at is the runtime routine. You probably remember that the 'if **name** == "**main**"' runs if we are calling the program as a standalone rather than as a library. The next two lines are the dummy data that I used to test the program. You could comment out the first one and run it with the second data line which is the one that uses the tuple. The last three lines will instantiate the wxPython routines, then the Line class and finally call the app.MainLoop wxPython routine to get the frame to run.

```
if __name__ == "__main__":
    data = (300,20,47,96,1200,700,500,230,179,500,300,20,47,96,200,400,500,
```

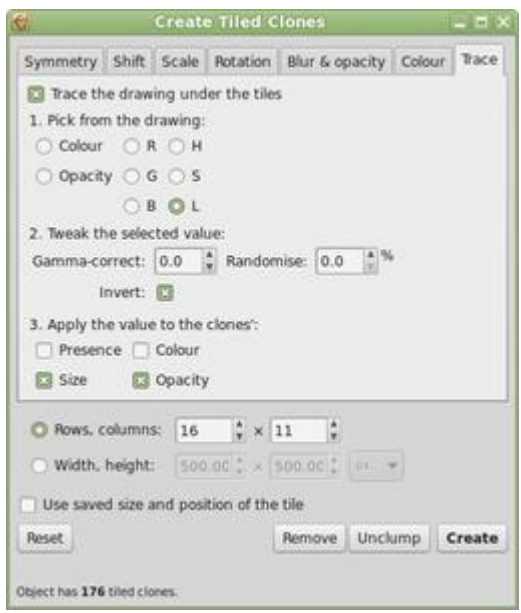
```
#data = (("02/01/2015",169.63),("02/02/2015",188.81),("02/03/2015",61.8

app = wx.App()
Line(None, -1, 'Bar Chart',data,"Monthly Sales - Colorado Springs")
app.MainLoop()
```

So there it is. Our own graphing/charting program and library. I've put the full code up on pastebin at http://pastebin.com/m2feeh5P. Until next time, have fun coding.

Drawing with Inkscape – Part 36

By Mark Crutch

The last tab of the Tiled Clones dialog is labelled “Trace” and, unsurprisingly, it's used for tracing existing drawings or bitmap images to create an array of clones that are related to some aspect of the underlying image. Initially it looks complex, but it's actually far simpler than it appears.



After using the Reset button in the dialog, most of the fields in this tab become disabled. To activate tracing mode, you have to first check the box on the top line. From then on it's just a case of choosing the options you want from each of the three numbered sections. These can be broadly described as:

1. Which aspect of the image do you want to use as your source of data values?
2. How do you want to modify those values?

3. What features of the clones do you want those values to apply to? .jpg) To demonstrate, I'll use the familiar face of La Giaconda as a background image, with a red round-cornered square as the selected parent object for cloning. When tracing, the clones are first placed, sized and rotated in accordance with the other tabs in the dialog, then the background image is tested underneath each clone's position to extract a value that will be used to further adjust the clone.

In section 1, we need to tell Inkscape what value to extract. You can choose from a general color value or one of the individual components of that color – Red, Green, Blue, Hue, Saturation or Lightness. There's also an option for Opacity, which extracts the alpha value from the underlying image. This section results in a single number for each clone. If “L” was chosen, for example, the dark parts of the image will yield a low number, whereas the light parts yield a high number.

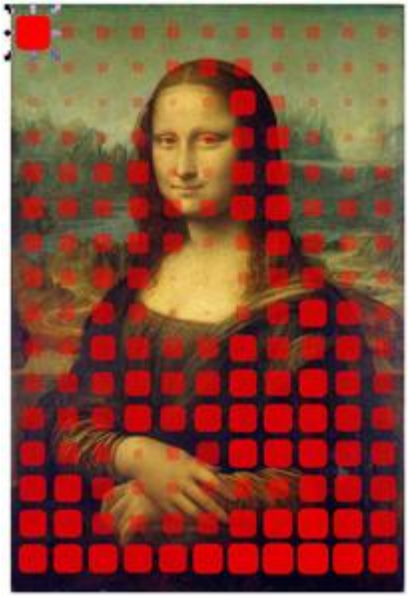
Section 2 gives you the opportunity to modify the collection of numbers that you've created in section 1. You can adjust the “gamma”, which allows you to shift the midpoint of the number range. This is best adjusted in small amounts once you've seen what the output looks like. Positive numbers will shift the mid-point upwards, negative values will shift it downwards. For example, when tracing a dark image using the “L” channel, you might want to shift the mid-point down – to better center the range within the values you're extracting. When tracing a light image, on the other hand, you might want to shift it upwards.

The Randomise field, as you might expect, just modifies each value from section 1 by a random number. The value in this field represents the maximum amount of randomness that is allowed. The Invert checkbox swaps

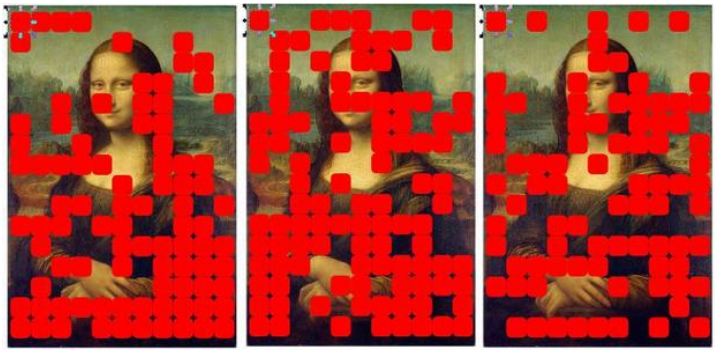
low values for high values, and vice versa. It's used when you want low numbers from your light areas, and high numbers from the dark areas, for example.

Finally, section 3 is where we tell Inkscape what to do with the extracted, adjusted values. For each clone, you can use the corresponding number to modify any combination of Presence (the probability that the clone is created or not), Colour, Size and Opacity. The latter two are the easiest to explain: in both cases the original parent element represents the maximum value that is allowed. None of your clones can get any larger, or more opaque, than the parent. Instead, the generated values are used to reduce the size of the clone, or reduce its opacity (making it more transparent).

Let's take a look at an example. I've traced the Lightness values of La Giaconda, adjusting them with a Gamma of -1.0, and finally applying the results to the Size and Opacity of the clones. The parent object is in the top left, and you can see that dark areas (where “L” is low) result in small reductions of size and opacity. Light areas (where “L” is large) result in much bigger reductions. Checking the Invert control would have resulted in the opposite effect: large, opaque clones in the light areas and small, transparent ones in the dark areas.



The Presence option in section 3 is, in my opinion, broken. A sensible way for this to work would be for values above a certain threshold to result in a clone being drawn, with values below that threshold producing no clone. That's not what happens though. Rather the value is used to adjust the probability that a clone will be drawn. Whether it actually is drawn or not then comes down to (weighted) luck. With “L” selected in section 1, it becomes likely that clones will be drawn in dark areas, and less likely they'll be drawn in light areas, but as it's all down to chance, it makes the results somewhat unpredictable. To make matters worse, the presence calculation, in all its randomness, is performed every time you click the “Create” button. Click it again and the arrangement will be different. Another click, another arrangement. Just take a look at this example – all three images were taken using exactly the same settings, just by clicking the button three times. With such radically different results from the same input values, it's no wonder I question the benefit of this “feature”.



The Colour checkbox is perhaps the most useful of all the options in section 3. To use it you first have to unset the fill of your parent object, just as we did last time when using the “Colour” tab. This option is a little different to the others because it doesn't use the value from section 1. Instead, it takes the RGB value of the pixels under the clone and, after adjusting them in accordance with section 2, applies the resultant color to the clone. A picture describes it far better than a lot of words.

This effect usually works best with a large number of

fairly small clones. That combination can bring a lesser computer to its knees fairly quickly, though, so be careful before you hit the Create button.

One limitation of this dialog is that each of the four options in section 3 is tied to a single set of manipulations in section 2. You can't have Opacity set directly by the “L” value, whilst also having Size modified by the inverted value, or use a different gamma for the Colour and Presence. It's not usually a problem, but if you do need such combinations then you've little choice but to use the Tiled Clones dialog as a starting point, then manually adjust each clone afterwards.

The Trace tab can be used in conjunction with the other tabs if you want to add some rotation to your clones, or perhaps adjust their size or spacing. The latter can be especially useful if you need to tighten up the array when using the Size checkbox in section 3. You should also bear in mind that the examples I've given here are deliberately simplistic. Your parent object can be almost any shape or group. The background you're tracing, similarly, can be a collection of objects rather than a bitmap image. Just remember that cloning complex parents takes more processing power and memory, so prepare for slowdowns or even crashes if you push the complexity or number of clones too high.

To conclude our look at the Tiled Clones dialog, I'm going to return to a couple of controls that I skipped over previously. The first is the “Use saved size and position of the tile” checkbox at the bottom of the dialog. Remember how Inkscape uses the bounding box of the parent as the basic unit of measurement throughout this dialog? This checkbox overrides that default to instead use the “saved” bounding box dimensions – where “saved” just means “the last value used before you checked this box”.

Consider our color trace of La Giaconda. By using 16 rows and 11 columns, my particular rounded square covers the space quite nicely. But what if I decide to rotate the square to make a diamond? Doing that increases the size of the bounding box, so my 16 rows and 11 columns now spills well over the edges of the background image.

One obvious solution is to reduce just the number of rows and columns, but that still leaves me with big gaps between the tiles. I could tweak the values in the Shift tab to compensate, but as I already had a tile placement that suited my needs, it's easier just to tell Inkscape to “save” that and re-use it for subsequent tiled clones.

Returning to the image as it was previously (by pressing CTRL-Z a few times) gets me back to a tile layout I'm happy with. At this point I can tick the “Use saved size...” checkbox to use that layout even while I change other parameters in the dialog, or make modifications that will alter the size of the parent's bounding box. Rotating the parent again, and adding a little variation with the

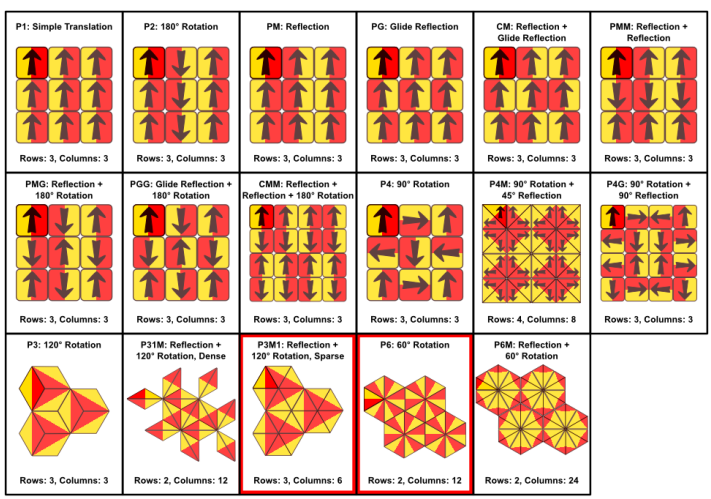
Randomise field in the Rotation tab, produces a much better result.

The last control to cover in the Tiled Clones dialog is actually the first one. It's the pop-up menu on the Symmetry tab, which determines how the tiles are initially arranged, prior to any adjustments from the other tabs. Mathematically speaking, there are 17 possible symmetries for a two-dimensional pattern – see the Wikipedia page for “Wallpaper group” for more technical details. These symmetries are based on translating (moving), rotating and reflecting the parent object to create a repeating pattern, and the pop-up lists each symmetry by its mathematical title in “short” notation (such as “P1”) followed by a description of the symmetry (such as “simple translation”).

So far, I've only used P1 for this tutorial, because it's the simplest to understand when used with the other tabs in the dialog. Other symmetries can quickly lead to complex (and confusing) results, but they can be useful if you want

to create tiled patterns or kaleidoscopic art. Some work best with triangular arrangements of objects, others with more rectangular parents.

Unfortunately, however, it's very easy to confuse Inkscape when using more complex symmetries. Even something as simple as switching your document from portrait to landscape mode can result in wildly misplaced clones. The following page shows examples of all 17 symmetries, but several of these had to be created in a fresh document and imported into the final image in order for them to work as expected. In a couple of cases, I was never able to produce the expected arrangement. They came close, but I had to manually arrange the clones a little to get the correct result. I've outlined those ones in red. If you do want to use the advanced symmetries, I recommend checking out this page in the official manual, which contains a similar set of images but with a little more detail: http://tavmjong.free.fr/INKSCAPE/MANUAL/html/Tiles-Symmetries.html


P1: Simple Translation
Rows: 3, Columns: 3
P2: 180° Rotation
Rows: 3, Columns: 3
PM: Reflection
Rows: 3, Columns: 3
PG: Glide Reflection
Rows: 3, Columns: 3
CM: Reflection + Glide Reflection
Rows: 3, Columns: 3
PMM: Reflection + Reflection
Rows: 3, Columns: 3
PMG: Reflection + 180° Rotation
Rows: 3, Columns: 3
PGG: Glide Reflection + 180° Rotation
Rows: 3, Columns: 3
CMM: Reflection + Reflection + 180° Rotation
Rows: 3, Columns: 3
P4: 90° Rotation
Rows: 3, Columns: 3
P4M: 90° Rotation + 45° Reflection
Rows: 4, Columns: 8
P4G: 90° Rotation + 90° Reflection
Rows: 3, Columns: 3
P3: 120° Rotation
Rows: 3, Columns: 3
P31M: Reflection + 120° Rotation, Dense
Rows: 2, Columns: 12
P3M1: Reflection + 120° Rotation, Sparse
Rows: 3, Columns: 6
P6: 60° Rotation
Rows: 2, Columns: 12
P6M: Reflection + 60° Rotation
Rows: 2, Columns: 24

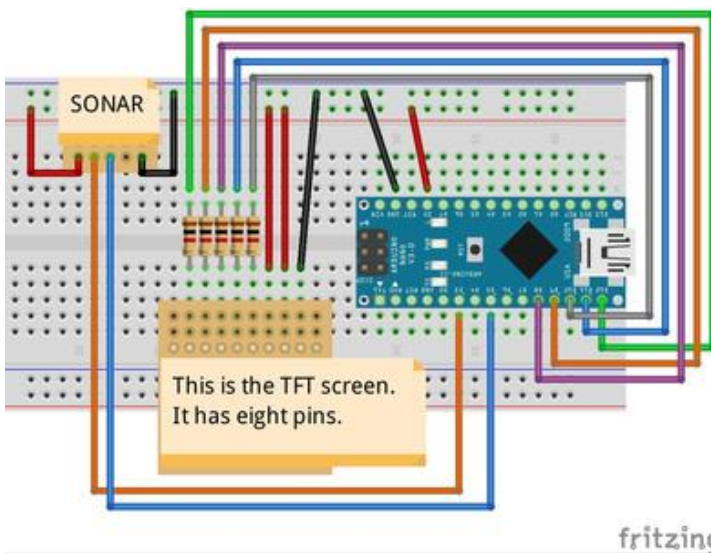
Arduino

by Ronnie Tucker

So, we'll carry on from last month and add the sonar sensor to the circuit.

Again, these are pretty cheap, but also pretty powerful, and, as we'll see, pretty accurate too.

The sensor I have has five pins. Some have only four, as, even with mine, only four are used. One pin is for 5V with one for GND. The other two are for echo and trigger. Mine is wired up as I've indicated in the diagram.



With the code from last month (http://pastebin.com/UYQe58xB), we can now detect a distance and display it on the colour screen.

You can see a video of it in action here: https://www.youtube.com/watch?v=Tbg-qyeJM6U. As you can see from

the ruler (in the video) it's pretty accurate!

Let me run through a couple of lines of code with notes and thoughts:

```
#include <NewPing.h>
```
Some of the echo and trigger stuff can be tricky, but with the NewPing library, t

```
#define TRIGGER_PIN  5
#define ECHO_PIN     3
#define MAX_DISTANCE 200
```

The three defines are pretty self-explanatory. Three and five are the Arduino pins, and the maximum distance is in centimeters. Some estimations of these units have them hitting a target over 400cm away!

In the main loop():

Make the integer uS become equal to the sonar ping in centimeters. There is also a sonar.ping_in() command for inches.

```
int uS = sonar.ping_cm();
```

We set the colour (as we did last month) but here we switch fonts:

```
myGLCD.setFont(SevenSegNumFont);
```

Maybe this one caught you out last month:

```
myGLCD.print(String(uS)+String(" "), LEFT, 24);
```

This one took me a bit to figure out. When I was using:

```
myGLCD.print(uS, LEFT, 24);
```

I was getting all kinds of string errors. Turns out it's (apparently) best to convert the integer to a string before

printing it, so the line is now:

```
myGLCD.print(String(uS)+String(" "), LEFT, 24);
```

This prints the number in uS (but as a string) to the screen, puts a space after it, and prints it aligned left at line 24. Why a space? Well, when you run it, you'll see that, without the space, it leaves the old number behind and you can end up going from 13 to 93. It actually means 9 but has only overwritten the 1 from the 13 leaving the three behind. It's a bad botch, I admit, but that's why I'm an amateur at this!

Programming JavaScript – Part 1

By Gergo Bogdan

You could ask why I chose to present JavaScript in this series of articles. The response is simple, JavaScript is the most trending and the most significant growth language in the past few years; there are very many open source projects which are using it, and you can find a lot of tutorials and guides on the Internet to learn it. Nowadays, you can develop anything in JavaScript; it does not matter if we are talking about web pages, single page applications (SPA), mobile applications for Android, iOS and Windows Phone, or IoT (http://en.wikipedia.org/wiki/Internet_of_Things) devices. Even a Linux emulator (http://bellard.org/jslinux) was written in JavaScript.

It is well known that JavaScript (usually) runs in the browser, but not only. Through the years, each browser had its own JavaScript engine and runtime; there was no standard for the language which the browsers could implement. In many cases, the difference was not only in the JavaScript engine, but in other functions of the browsers, like how these displayed content on the page, how was the styling applied, and so on. Because of these differences, if a developer wanted to address users worldwide, he had to support all the browsers, write the JavaScript code for each browser, test it, maintain it, etc.

The segregation of JavaScript implementations did not help the language to get embraced by the community and web developers. This was the situation until 2006, when the first stable version of the jQuery (http://jquery.com) JavaScript library appeared. This library helped developers to support different browsers, because it offered a unique way to apply JavaScript logic.

Nowadays, the three most widely used browsers (Google Chrome, Firefox and Internet Explorer) got to a point when these can support the standard of the language, so developers and users have a more unified programming and running environment.

The first version of JavaScript was developed by Brendan Eich (http://en.wikipedia.org/wiki/Brendan_Eich); at that time he was working at Netscape (now it's called Mozilla). During the development phase, the language was called Mocha, but when they officially released it in 1995, it was called LiveScript and afterwards when the Netscape Navigator browser (ancestor of Firefox) 2.03B version was released, they renamed the language to JavaScript.

In the past twenty years, there were different versions of the language and there was a need to define a standard. The standardization was done by ECMA International and the standard for the language is called ECMAScript (http://www.ecma-international.org/memento/TC39.htm) During the years, each platform and browser had its own version of JavaScript; for example Microsoft had released their version named JScript. In the meantime, Netscape was starting to use JavaScript on the server side too (SSJS – server side JavaScript). Nowadays, node.js is the de facto for server side development using JavaScript.

As the language's name suggests, this is a script language; it needs a runtime (engine) to be executed. We call this the JavaScript engine, and these runtime environments (try to) implement the ECMA standard. The most widely known and used JavaScript engine is V8 (https://code.google.com/p/v8/), SpiderMonkey (https://developer.mozilla.org/en-US/docs/Mozilla/Projects/SpiderMonkey), Rhino (http://en.wikipedia.org/wiki/Rhino_(JavaScript_engine)), Nitro, and Chakra (http://en.wikipedia.org/wiki/Chakra_(JScript_engine)). V8 was developed by Google, SpiderMonkey and Rhino are supported by Mozilla, Nitro

is backed by Apple, and Chakra is developed by Microsoft. Today, V8 is the fastest JavaScript engine.

Nowadays, there is no webpage which would not use some kind of JavaScript for displaying content, modifying the layout, loading data, displaying charts, generating PDF files or even editing images. So, in case you want to build web pages or build your own web site, learning a little bit of JavaScript (JS) will help to do the job.

The purpose of this series is to present the JavaScript language using simple examples. In the beginning we will need only a text editor, it's good if it has syntax highlighting. It can be Notepad++, Gedit, Vim, Emacs, but you may use anything you like. Syntax highlighting helps to visually separate different parts of the code (keywords, values, comments, types) and spot coding errors quickly. We will need a browser too. I prefer Google Chrome, but you are free to use any browser; the examples in the beginning will run in the same way in every browser. The JS's syntax is C-based, and it supports object oriented (http://en.wikipedia.org/wiki/Object-oriented_programming), functional (http://en.wikipedia.org/wiki/Functional_programming) and imperative (http://en.wikipedia.org/wiki/Imperative_programming) programming paradigms.

In the beginning, we will focus only on client side JS, so we will need a minimalistic web page to run our JavaScript. On the client side, besides JavaScript, other technologies, like HTML, CSS, XML, JSON are used too; we will talk about these later.

Let's start the series with the traditional hello world program. First we create a new JavaScript file, let's name it hello_world.js. The extension of JavaScript files is .js. Content:

```
//everything after the // characters will be ignored by the JavaScript engine, th
//within your code
alert("Hello JavaScript World!");
```

Now, we need to create the html webpage, let's name the file hello.html. Content:

```
<!--HTML Comments should be put between these symbols -->
<!DOCTYPE html>
<html>
   <!--The header part of an HTML document -->
   <head>
   <!--It is a good practice to specify the character set used on the webpage. -->
   <meta charset="utf-8">
   <!--Lets include the hello_world.js file which was created before -->
   <script src="hello_world.js" type="text/javascript"></script>
   </head>
   <body>
   <!--Currently the webpage content is empty -->
   </body>
</html>
```

HTML is a markup language which is used in web development. As we advance with the series, I will present more and more HTML elements besides JavaScript programming.

The HTML files contain tags (also called HTML elements), which are present in the form of &lt:element_name>&lt:/element_name>. These tags are interpreted and rendered by the browser and display the content on our webpages.

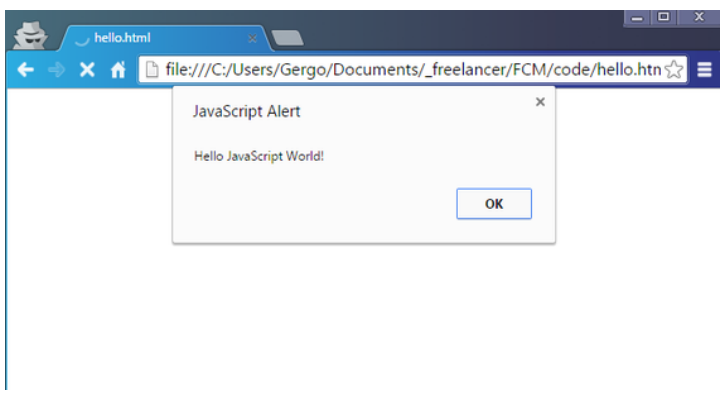
HTML files have three main parts:

1. HTML File version – this is the first line in the HTML web page, usually the version is between &lt:!...> symbols. The current version of HTML is HTML5, which has a very simple version notation: &lt:!DOCTYPE html>. Although browsers usually do not complain if this tag is missing, it is a good practice

to add it as the first line in our webpage.

2. HTML head element (&lt:head>…&lt:/head>), this element is special, because the browser uses it to gather important information about the webpage, like the title (can be added with &lt:title>…&lt:/title> tag), and meta data, like the charset. But there are other options like author: &lt:meta author="John Doe"> and many others. The meta tags also help search engines to get information about your website and rank it in the search results.
3. HTML body (&lt:body>…&lt:/body>) element, this is the content of the webpage, everything within the body tag is displayed to the user, we will cover more on this later.

We can run our Hello World application by opening the hello.html file in a browser and we should see something like this:



The execution flow in the browser is the following: when the browser loads the hello.html file, in the first line it can see that this is an HTML5 file, it reads the head element and, as specified in the meta tag, it loads an UTF-8 charset. The next line is the script tag: &lt:script src="hello_world.js" type="text/javascript">&lt:/script>. With the help of the &lt:script> html element, we can include external script files (like we did in this

case) or define new code blocks within the html file. As the browser loads the hello_world.js file, it executes the code line:

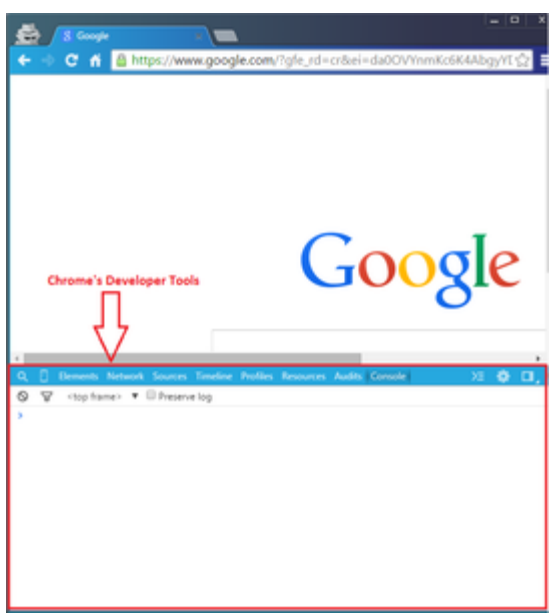
```
alert("Hello JavaScript World!");
```

The alert() function is available in every browser, it shows a pop-up window with the message as its parameter.

We have two other options to log or show messages to the user. One of the options is the console object; this prints values to the Console output of the Developer Tools within the browser. In that case the JavaScript code would be:

```
console.log("Hello JavaScript World!");
```

In Chrome, you can fire up the developer tools by pressing the F12 key.



In case you are using Firefox, you can install an add-on, called FireBug (https://addons.mozilla.org/en-us/firefox/addon/firebug/) which is an awesome tool for developers.

The second option to display messages to the user is by adding those to the html page using the document.write() method. I will leave this for you to explore.

In the next part, we will look at JavaScript objects and functions; how can these be created, what are they, and why are they good to use. In case you would like to read about a particular topic, please email me at: greg@grelution.com.

*Bio: My name is **Gergo Bogdan**, Igreg@grelution.com am a software engineer, blogger, tech enthusiast from Budapest, riding the waves of the constantly changing IT ocean ☺, you can check my website http://grelution.com.*

Using Latex – Part 2

by John Kerr

By the time you may be reading this, many students will be wrapping up their school year. The summer break is a great time to learn LaTeX. It is said that LaTeX can have a steep learning curve. I agree sometimes, sometimes I do not, ultimately it all depends on what you want to do, and there are ways to level off the steep slope. Today's LaTeX beginner at least has YouTube videos about LaTeX to help reduce the slope. Also, the Internet has many free e-books and tutorials to learn from.

Build Your Reference Library

In your home directory I suggest that you create a directory to hold your Latex documentation files. It can be anywhere in your home directory. To get you started, here is a short list of resources that I have found helpful:

Five Minute Guide To LaTeX. Horst & Wennecker. http://www.howtotex.com/download/FiveMinuteGuideToLaTeX.pdf

LaTeX Tutorials: A Primer. Indian Tex Users Group. www.tug.org/twg/mactex/tutorials/ltxprimer-1.0.pdf

The Not So Short Introduction To LaTeX2e. Tobias Oetiker. https://tobi.oetiker.ch/lshort/lshort.pdf

Let's Learn LaTeX. S. Parthasarathy http://drpartha.org.in/profpartha/teachlatex.htm

Beamer by Example. A. Mertz and W. Slough. http://www.tug.org/pracjourn/2005-4/mertz/mertz.pdf

You may want to have a copy of this directory on your

tablet (if you have one). However, the purpose of this library is to provide you with much more detail about LaTeX than I can provide here, and to give you code that you can cut and paste into a blank document using one of the LaTeX text editors that you have installed. This is a quick way to discover how the whole shebang fits together and works.

You may find that you are never finished building your reference library. Searching the Internet will find thousands of files to help you with the type of document you are creating.

Build Your Template Library

Not quite as important as your reference library, but, as you search for LaTeX information, you are bound to find some templates. The Internet has plenty of document templates that you can download to kick-start your projects and/or to get an understanding of LaTeX. Place each template in its own directory and when you use a template, copy the contents of the template directory into a new working directory.

The dual process of reading instructional texts and studying the source code of templates has fast-tracked what I have been able to accomplish with LaTeX.

For instance, the task of creating a poster from scratch would take a very long time, but why do this when there are so many good templates that LaTeX users are happy to share? I downloaded a few templates of posters before I settled on the one that I created a poster with. All I had to do was replace the text and pictures in the template with my own.

There are as many templates to download as there are document types: letter, slide, report, book, poster, newsletter, thesis.

The part of a LaTeX file that handles the type of document to be created is called the preamble:

documentclass[12pt,letterpaper]{article}

The above line is the heart of the preamble, setting up the document to use 12pt font on letter-paper, and it is an article.

There is other code that follows within the preamble, but for the sake of brevity we will leave it at that for now.

The preamble is followed by:

\begin{document}

Where you type in your knowledge with formatting.

The end of the document is tagged with

\end {document}

Some LaTeX editors have wizards that help you create your preamble. In time, you may want to create your own template with the components in the preamble that you need most often.

With your reference library, Template Library, and the information here, you are now ready to create beautiful documents with LaTeX.

A good program to experiment with LaTeX is called Gummi.

The program always opens up to the welcome document. Save this under a new name and then edit the text.

For instance, place an * at the end of the "\section" so it looks like "\section*". Section one loses its number and section 2 becomes section 1. LaTeX keeps track of this sort of enumeration. You can insert an extra section, and then, when the document is recompiled (hit F9), everything is updated with new numbers.

Highlight text and press the "B" icon to bold the text.

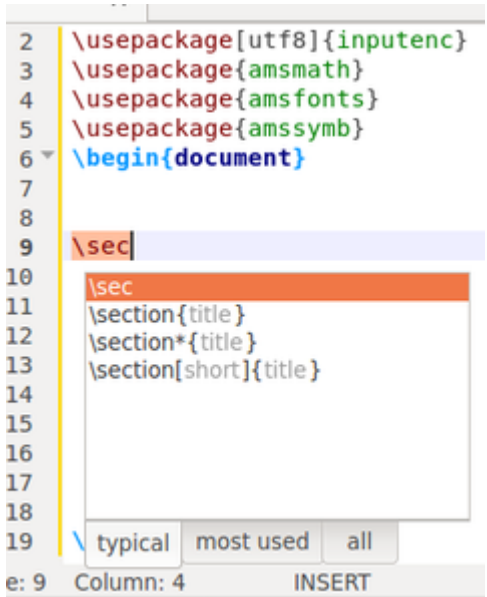
F9 is the standard key in many editors to compile a document in LaTeX. Many of the commands that you use with the ctl-key in other programs work the same way with LaTeX editors. Compile is not the same as save. I always save before I compile.

Gummi has its shortfalls

What Gummi lacks is a robust hierarchical menu system found in other editors that lists the types of LaTeX code

that you would use.

Gummi also lacks an autofill feature that provides examples of LaTeX code by typing the first few letters. Most LaTeX editors have this feature.



I do prefer to use Gummi when I want to type up a simple page or report. The other times I use TexStudio.

Are you feeling overwhelmed by LaTeX? Don't. Today I find LaTeX very easy to use, yet I started using LaTeX as a production tool a very short time ago. Give it time.

So far, we have discussed how to set up your computer to work with LaTeX. Specific “How To” articles are to come. However if you have been following along, I bet you have been doing some pretty neat stuff already.

Next issue we will grab a template from someplace and have some fun.

SPIDEROAK
.com
Online BACKUP
Secure SYNC
Easy SHARING
Whether you need to access a document you have stored on a remote server, synchronize data between a Mac, Windows or Linux device, share important business documents with your clients, or just rest easy knowing all of your data is safely, securely, and automatically backed up - SpiderOak's free online backup, online sync and online sharing solution can handle all your needs!
SpiderOak offers a different approach to online backup by combining a suite of services into one consolidated tool - free online backup, synchronization, sharing, remote access, and storage. This difference is further measured in our zero-knowledge privacy policy - the first one ever employed in this setting. Our flexible design allows you to handle data from any operating system (Mac, Windows and Linux) or location (external drives, network volumes, USB keys, etc...) using just one centralized account.
Access all your data in one de-duplicated location
Configurable multi-platform synchronization
Preserve all historical versions & deleted files
Share folders instantly in web ShareRooms w / RSS
Retrieve files from any internet-connected device
Comprehensive 'zero-knowledge' data encryption
2 GBs Free / $10 per 100 GBs / Unlimited devices
https://spideroak.com
Download mobile clients for iOS & Android
JOIN SPIDEROAK NOW
Get 2 Free GBs
Get 25% off any SpiderOak package with the code: FullcirclemagFans

Write For Full Circle Magazine

Guidelines

The single rule for an article is that it must somehow be linked to Ubuntu or one of its many derivatives—Kubuntu, Xubuntu, Lubuntu, etc.

Write your article in whichever software you choose—I would recommend LibreOffice. But, **please spell and grammar-check it!**

The Official Full Circle Style Guide can be read at: http://url.fullcirclemagazine.org/75d471

Please read this document before submitting an article. Follow the guidelines and you will have a much better chance of seeing your article in Full Circle.

Writing

There is no word limit for articles, but be advised that long articles may be split across several issues. In your article, please indicate where you would like a particular image to be. Please do not use any formatting in your document.

Images

Images should be no wider than 800 pixels, in JPG format, and use low compression.

When you are ready to submit your article, please email it

to: articles@fullcirclemagazine.org

Non-English Writers

If your native language is not English, don't worry. Write your article, and one of the proofreaders will read it for you and correct any grammatical or spelling errors. Not only are you helping the magazine and the community, but we'll help you with your English!

FOR REVIEWS:

Games/Applications
When reviewing games/applications, please state clearly:
• title of the game
• who makes the game
• is it free, or a paid download?
• where to get it from (give download/homepage URL)
• is it Linux native, or did you use Wine?
• your marks out of five
• a summary – with positive and negative points

Hardware
When reviewing hardware, please state clearly:
• make and model of the hardware
• what category would you put this hardware into?
• any glitches that you had while using the hardware?
• easy to get the hardware working in Linux?
• did you have to use Windows drivers?
• marks out of five
• a summary – with positive and negative points

You don't need to be an expert to write an article – write about the games, applications and hardware that you use every day.

Ubuntu Phone

I'm hoping to keep a page or two each month exclusively for Ubuntu phone/Touch news. This month, though, Cristian Parrino (Vice President of Mobile at Canonical) answers some of the questions you folks sent in that I couldn't answer. -- Ronnie

Is it possible to SSH into the Ubuntu Phone as you would with desktop Ubuntu?

Yes, there are instructions for how to do this here: http://askubuntu.com/a/599041

Is it possible to use rsync for backing up the Ubuntu Phone?

This is not supported but backup services are planned to be introduced with partners.

Is there a root account on the Ubuntu Phone?

Yes, access to it is via sudo.

Will the Ubuntu Phone be able to run Android apps?

No, phones running Ubuntu are not able to run Android apps. Ubuntu supports apps written in HTML5, Qt/QML, and we also have our own Scopes toolkit. Scopes bring the content and services typically hidden behind apps, right to the screen - for a better experience for users and developers.

Any plans to bring WhatsApp to the Ubuntu Phone? (LOTS of people asked about that)

We aim to have all the key apps available on Ubuntu, in time. For WhatsApp specifically, they have the opportunity to use our messaging framework to integrate their service directly into the core messaging app on the phone - essentially becoming the equivalent of iMessage on iPhone, albeit with branding and theming. Much easier than porting an app to Ubuntu, and much more valuable.

Any word on being able to buy the phone outside of Europe?

We have announced the Meizu device - in addition to Europe; these will be launched in China as well. We're working hard to secure a device for the US market also, but have nothing to announce at this time.

Is Bluetooth supported with the Ubuntu Phone? The phone specifications suggest that it's not supported in the software.

Bluetooth is supported; however we do not currently support Bluetooth LE (low energy). This will be coming soon.

Will it be possible to install the Ubuntu Phone OS on Android devices?

It is already possible to install a version of Ubuntu phone on some Android devices (e.g. Nexus 4), and there are several community ports for other devices available online. These are suitable for a developer experience.

Can the Ubuntu Phone (or OS) run Ubuntu desktop apps on phones?

Not currently, but this is very much part of our roadmap.

Is there a plan to sell Ubuntu Phones on a permanent basis? (ie: not flash sales)

Yes, in fact the last flash sale has already taken place. Over the next few weeks, phones will be available directly from the BQ website.

Is the Ubuntu Phone compatible with MS Exchange (ie: for businesses)

MS Exchange is not supported.

Is there a plan for better integration with the various flavors of Ubuntu for syncing and backup?

Syncing and backup services will be provided by third parties (we shut down our own Ubuntu One sync service last year).

Will the Ubuntu Phone OS implement a 'kill switch' for lost devices?

This is a requirement that is being evaluated for development.

How good is the Ubuntu Phone on privacy/NSA/

encryption?

A topic that is dear to us. Ubuntu is historically known as a highly secure OS (http://www.zdnet.com/article/uks-security-branch-says-ubuntu-most-secure-end-user-os/) - on the phone, we've extended this to the app permissions framework. Users decide what to share with what app at a much more granular level than they do on other platforms.

Many thanks to Cristian for taking the time to answer those questions.

Latest Ubuntu phone OS updates:

OTA:

- Battery life improvements
- Screenshots no longer show the volume indicator
- Notifications can be dismissed by swiping
- Photos can be edited directly from the Camera photo roll
- Fullscreen and Normal modes are now supported in Gallery photo display
- Improved support for BT devices that require a Pin code for pairing

Telegram:

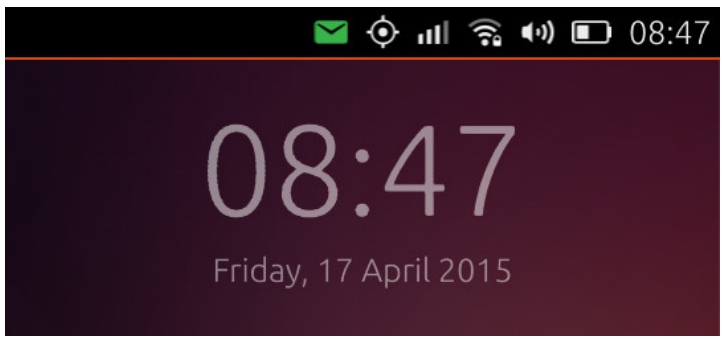
- Sending / receiving documents - no longer limited to just photos and videos
- Telegram activity on the lock screen - see your Telegram statistics on your lock screen (double tap the infographic to change the information displayed)
- Image previews are now zoomable

- Improved notifications support
- Added support for message forwarding
- Various other UI improvements and fixes

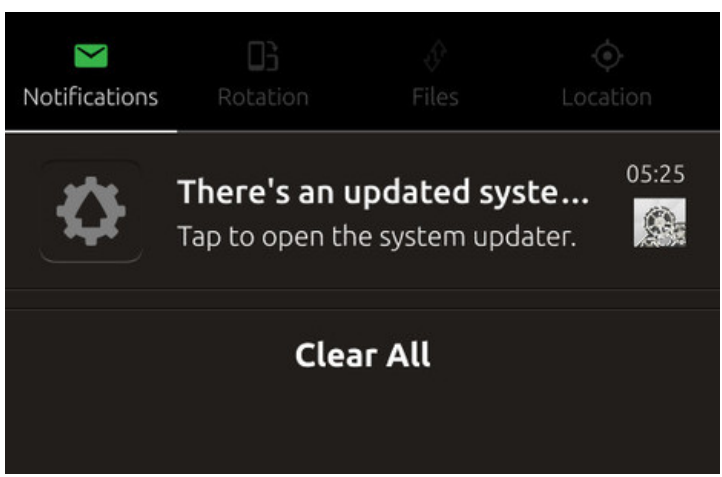
Ubuntu Phone Update OTA

An update OTA (Over The Air) is quick and painless with Ubuntu phones.

First, you'll get a notification (green envelope icon) at the top of the screen.

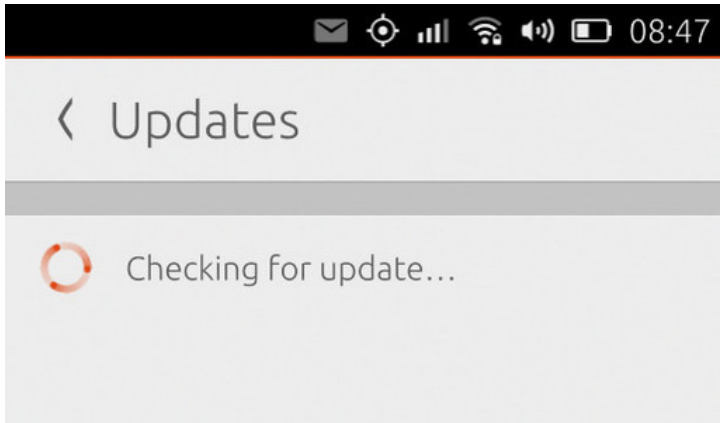


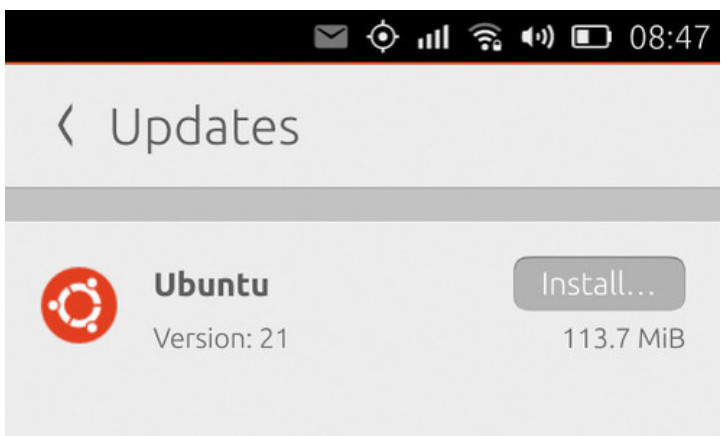
Pulling down on the notification bar will show you the message. In this case, a system update.



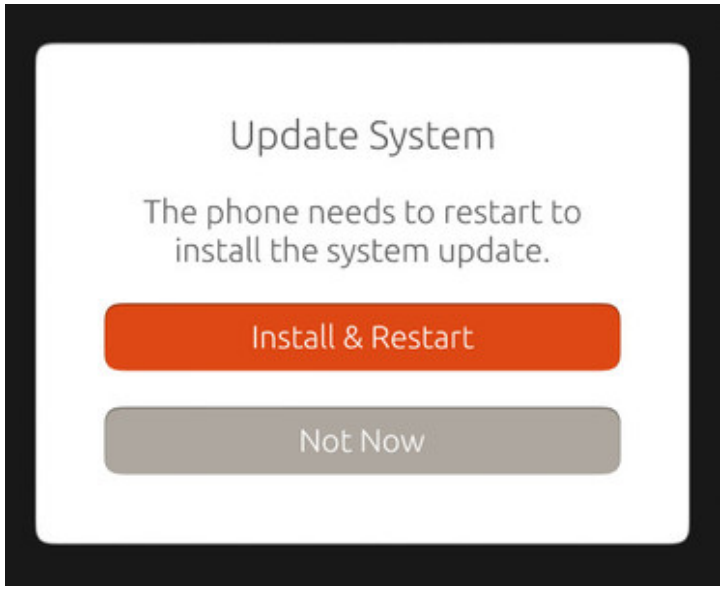
Pressing the System Settings icon (the cogs) on the right

will take you to the update screen where it will run a quick check.





On pressing the ‘Install…’ button you’ll need to restart the phone to let the update do it’s install.



Once the phone has gone through a reboot you have your newly updated OS!

08:51
OS Build Details
OS build number
21
Ubuntu Image part
20150410.1
Ubuntu build description
Ubuntu Utopic Unicorn (development branch) - armhf (20150410-232623)
Device Image part
20150408-4f14058
Device build description
KRILIN01A-S15A_BQ_L100EN_2021_150410
Customisation Image part
1428618470

LETTERS

Ubuntu Phone First Impressions

Regarding development of apps for the Ubuntu Phone, I found one particular link http://www.ubuntu.com/phone/developers

My first non-technical impressions on the phone :

- I like the overall clarity of the menus and the navigation inside the phone.
- I really miss some notifications: it should be clearer without having to go to the notifications menu (scope?).
- I think the security aspect, and private life respect from the apps is clearly what seems to make this OS different from android and IOS.
- I think ubuntu should write ubuntu store and the translation. In French, when I read magasin ubuntu, I was a little scared to find myself in a shop.
- Big problems with GPS.

One of my daughters who played with the phone says :

- The whole organization of the menus and the OS is really different from her android. She appreciates this aspect, especially the left menu and the fact there is no physical button.
- She would like to have more apps in the phone (in fact in the store).

My son enjoys the few games I installed for him. In fact, he loves this phone, the slidings from left to right, from right to left. He already knows how to use it, really faster than my Android phone (an HTC One X+)

Marc

TexStudio

In the FCM#95, I read your article about LaTeX. However, please accept my suggestion of talking (or at least referencing) TexStudio (http://texstudio.sourceforge.net/) as a good LaTeX editor.

I am really a regular user of LaTeX, and TexStudio was the best editor I ever found, with the great advantage of running in Windows, Mac, and Linux! I am surprised you did not reference it.

Tiago

What A Pig!

Regarding FCM95: On an old Medion Laptop AD 2003 with a P4 CPU and only 256MB RAM (later upgraded to 512MB which was a pig of a job having to strip part of the machine to get access to RAM), I was initially happy at using a USB WiFi stick with an Atheros chip. Since other WiFi sticks, made by Netgear and Linksys, and probably with a Broadcom chip, refused to work with Linux.

However since the release of (K)(L)Ubuntu 14.04, all the above WiFi USB sticks work in Linux without any configuration. It should mean that the Broadcom drivers are now included in the Linux kernel.

Frank

Chrome Cult

by S. J. Webb

Google's Chromebooks are becoming more common in the household. The Chromebook platform is a top seller for Amazon. Many schools and students use Chromebooks for education. A Chromebook is a simple, fast, and secure machine. It is supported by a strong tech giant that wants it to succeed. The Chrome OS has a cult following, and it is growing.

Why mention Chromebooks in this magazine? It is fairly common knowledge that Google's Chrome OS operates off the Linux Kernel. Let's take a brief history of the Chrome OS.

The Chrome OS is a refined and polished corporate product that is downstream from the open source Chromium OS project. This project is the raw developer mode to Chrome OS. I recommend visiting the Chromium website if you want further information. The Chromium developers recommend using Ubuntu 14.04 LTS as the starting point. Google points out that many of its developers use Ubuntu as the base to Chromium. However any Linux distro will work for developing. If you are not a developer, but would still like to help Google, you can be a beta tester for upcoming Chrome OS releases.

Additionally, Canonical was contracted in 2010 to help develop certain aspects to the Chrome OS.There is a list of open source projects that helped create the OS. The Chrome OS can be considered a streamlined Ubuntu derivative in some aspects.

There are core differences between Chromium and Chrome OS. The latter encompasses cloud computing. You

work within the Chrome Browser with various apps. The Chrome OS is optimized when it is connected to the internet. Chromium is a true desktop environment with access to full software programs; Linux packages are not Chrome OS compatible. The Chrome OS is found only on Google products. Chromium can be installed onto any laptop since it is open sourced. Chrome OS receives security and system updates, whereas Chromium does not. Anyone with a Gmail account can log in to a Chromebook, Chromium does not have this feature. It takes less than 7 seconds from login to the desktop.

Pic 1 is the Startup Screen

Pic 2 is the Login Screen

Pic3 is the desktop screen

There are limitations to a Chromebook, but the functionality trumps these issues. The Linux Kernel provides a strong base for Chrome OS. Various projects can be created using Google Docs, and some people even code with a Chromebook. I am not a coder, so I cannot comment on that aspect. However there are some items to ponder:

1. Each Chromebook has an end-of-life date of 4 or 5

years from date of OEM manufacture.
2. Some apps can be invasive.
3. Chrome OS is not a full desktop environment.
4. Not all peripheral devices are supported.
5. Cloud Printing can be a hassle to set up.
6. Cloud computing does not meet all of the demands of a full Linux Desktop Distro.
7. Google Web Store has limited number of apps.
8. No large internal storage options for data.
9. Long term Cloud Storage can be costly.

Functionality that helps promote Chromebook adoption:

1. Streamlined OS based off the Linux Kernel.
2. Supported by a large corporate entity.
3. Crouton can be enabled to use Ubuntu LTS 14.04.
4. Long battery life for daily use.
5. Fairly intuitive GUI.
6. Chrome OS can utilize Android Apps.
7. Easy software updates.
8. Built in-virus protection

I previously mentioned my Toshiba Chromebook 2 is a backup to my openSUSE HP 15 Notebook. The retail price on both laptops hovers at $250.00. A quick comparison between these two laptops illustrates the generalized hardware differences. See the table below.

	HP 15 Notebook	Toshiba Chromebook 2
Weight	5 lbs.	3 lbs.
Size	15 inches	13 inches
Battery Life	5 hrs.	7 + hrs.
Processor	Intel Pentium 2.66	Intel Celeron 2.16

Ram	4	2
Hard Drive Type	500 GB SATA	16 GB SSD
SD Card Reader	Yes	Yes
2.0 USB Ports	2	1
3.0 USB Ports	1	1
Ethernet/HDMI Ports	Yes/Yes	No/Yes
Wifi Capable	Yes	Yes
Audio	DTS Studio	Skull Candy
OS	Originally Win 8.1 OS; replaced with openSUSE 13.2 OS	Chrome OS

Since the Toshiba Chromebook is light on hardware I suggest owning the following items. There are 4 peripheral devices recommended for any Chromebook.

1. USB Hub to expand the number of USB slots.
2. USB Ethernet adapter in case Wifi is not available.
3. Wireless mouse to prevent touchpad frustration for the end user.
4. External DVD-Rom Drive.
5. External storage using a USB Thumb Drive or a SD card.

General Observations

The Chromebook is fast, but it takes some willingness to learn the “Chrome Workflow.” openSUSE recognized my iPhone via USB and Bluetooth in every attempt to connect. My iPhone does not finalize the Bluetooth pairing to the Toshiba. Both rigs can tether to the iPhone extremely well. I wonder if an Android smartphone would work better with the Chromebook? Using Google Docs was not too bad, but I still like LibreOffice.

Some excellent Chrome-dedicated Resources:

1. www.chromium.org
2. www.practicalchrome.com now known as www.chromebooks.today.com
3. www.omgchrome.com
4. www.gigaom.com search for Chrome Show

Future submissions will look into using Google Docs, Google Drive, Crouton, and various items pertaining to the Chrome OS.

Sources:

1. www.chromium.org
2. www.hp.com
3. www.toshiba.com
4. www.wikipedia.com

SJ Webb *is a Linux Hobbyist and Research Coordinator. He enjoys fishing, hot rodding, and spending time with his kids and wife. He thanks* ***Mike Ferarri*** *for his mentorship.*

My Opinion

Ubuntu phone – some thoughts from a paying customer

by Chris Burmajster

Well, we have waited nearly two years for the Ubuntu phone, and now it has finally arrived. Here are some of my thoughts and observations.

Firstly, the buying experience. People often talk about the experience of ownership, but you have to buy the product before you can own it. If the buying experience is poor, you can put that customer off buying one of your products again, assuming that they don't cancel the order out of sheer frustration!

And so it was that, on February 11th, I got to work quite early and quickly checked my home e-mails. Just as well I did, as there was an e-mail from Canonical saying that the Ubuntu phone was finally available to buy, but only for 12 hours – why? What on earth was that about? The fact that the phone was finally available to buy made me quite excited, as, like other Ubuntu fans, I have waited patiently and eagerly for this phone to appear.

At this point I did something I've never done before – I was prepared to buy a product I had never heard of, from a manufacturer I had never heard of. This is not me at all; I normally research my purchases carefully, reading reviews and comparing deals before parting with my hard-earned cash. I was taking Ubuntu and Mark Shuttleworth on trust.

I have used Ubuntu since almost the very beginning –

7.04. When they introduced Unity, I didn't like it at first but I stuck with it and now I'm very comfortable with it (with hindsight, I think that they introduced it before it was really ready). With the phone, and the whole idea of convergence between various devices, Unity makes sense. Obviously Shuttleworth was thinking ahead – a man with a vision. Also, Shuttleworth had given millions of people, including myself, a really great operating system, and all for free. I decided to step out in faith and buy the phone, sight unseen, and hope that it was as good as I hoped that it would be. I clicked the link and got taken to the BQ website and to the new Ubuntu phone. The excitement quickly turned into frustration and then anger.

I looked for the 'add to cart' button but unfortunately, it was missing. I refreshed the page many times, but to no avail. I tried another browser with mounting frustration, but got the same problem. What's the point of telling us that we can buy the phone if we can't buy the phone? After about 30 minutes, I refreshed the page again and finally got an 'add to cart' button. I clicked it and got a 502 error! I tried again and again - same thing. I clicked on the 'contact us' button to tell BQ of the problem but got a 404 error for my trouble! Left it for another 30 minutes and tried again. This time it worked, and I got another page forward, but when I clicked the 'continue' button it crashed again. This time I e-mailed Canonical to tell them of the problem, but got no answer.

My frustration was slowly turning to anger. What should have been a pleasure was turning into a nightmare. I left it for another 30 minutes and tried once again. This time I got to the payment page, filled out all the details and when I pressed 'send' - guess what? I left it for another 30 minutes and tried again. This time - FINALLY - it all worked and I was able to order the phone. It claimed that it was in stock, yet delivery is not until March - what's that all about? BQ charged 20 Euros for delivery; for that

sum I expect next-day delivery, not next-month delivery! However, as I ordered on the first day, they gave me a case for the phone which cost 16 Euros at no extra charge, so I shouldn't complain too much.

So the buying experience was crap. I'm reasonably patient, but to wait a month or more for delivery was really annoying, especially as I had waited nearly 2 years for the phone to appear. They should have had the phones ready to go to all those eager customers instead of making us wait. What they should have done is to say that it was a pre-order. That would have made it clear that the phone was not available for immediate delivery, and not raised expectations unnecessarily.

What was the point of the 'flash' sale? To make people rush to buy it? To generate some column inches about the numbers trying to buy one? This was a big mistake, in my opinion. If you're going to sell it, then sell it! After all, the future of the Ubuntu phone will no doubt depend on how well it sells and if you artificially restrict the sales, you'll just put people off. Ubuntu fans will slog through most problems, but the general public will not, and it needs to sell well to them to be a success. I wonder how many people who also had trouble with the BQ website just gave up? Also, many who go to the website, and discover that they can't buy it there, and then simply won't come back, and the Ubuntu phone will lose sales. A big mistake, in my opinion.

Hopefully, lessons will be learned for future Ubuntu product releases.

After having ordered the phone, I then tried to find out about what I had ordered – a classic case of putting the cart before the horse! There was next to nothing on the web, but, over the next few days, a few reviews and videos went online, and I was able to find out a bit more

about it. The first disappointment was that it won't function as a computer to which you can just connect a display, keyboard and mouse. That's a big shame. The specs weren't terribly good either. It's a pity that there was no choice, that there was only just the one phone available. I would have gladly paid more for a better specc'ed handset, but I understand that this is all a toe-in-the-water for Ubuntu. Hopefully, that will come in the future.

The lack of reviews was also a bit of a worry for another reason. I'm greatly concerned about privacy and the Snowden revelations horrify me, as do the way various apps grab your personal information and phone home with it, not to mention the way companies like Google track you around the web and flog that information for 30 pieces of silver to the highest bidder. Will the Ubuntu phone stop that? How good is it from a privacy perspective? How safe and secure is it? I hope that somebody will test it thoroughly from this point of view.

Anyway, to the phone itself. Unlike those lucky journalists – who got the phone weeks before us paying customers, and got it in a fancy box with a nice pair of headphones included – us paying customers just got a small box with a phone in it. The phone looks good and well made, but the protective case does not. It looks cheap and it's slightly too big, which means that the lid won't close neatly on top of the phone. Given that it sells for 16 Euros, it's poor value for money and I'm glad that I didn't have to pay for it.

The phone's biggest problem so far is that I cannot get it to load my phone book off the SIM card. There doesn't appear to be a 'import from SIM' option in the Contacts scope. Before taking the SIM out of the old phone, I carefully saved all my contacts onto the SIM card for easy transfer; shame that the phone doesn't appear to allow me

to import them. Surely we shouldn't be having this problem in 2015? Without a working phone book, I can't make any calls as, like most people, I can't remember everybody's number. They need to sort this out quickly. At present, I've had to go back to my old phone as at least it works. After a few days, I bit the bullet and started manually importing my contacts, one by one. The one advantage of this is that it enabled me to prune my phone book in the process.

Initially, although I was able to bluetooth the phone to the car, there were no phone books to import and even if I dialled a number manually, it didn't work. This was another disappointment. However manually populating my phone book, making a few test calls, sending a few test texts and switching bluetooth off and back on again, got it working.

The major problem with bluetooth is that you can only switch it on and off, there is no sub switch for turning off discoverability, as I have had in previous phones. This means that, if you want to use bluetooth you have to put up with the phone constantly trying to pair with anything in range, which must be both a security issue and it must also drain the battery somewhat.

The instructions say that the only e-mail scope available is for Gmail and that another scope is in preparation for the rest of us who don't have a Gmail account. I hope that we won't have to wait too long for that.

My final thought is about support. When you buy a phone from a shop or from a carrier, you can get help from them if you have questions, or when things go wrong. However, with the Ubuntu Phone, where is the help? I feel quite on my own. Of course, I can go on the ever helpful Ubuntu forums, but as I'm one of the first people to get an Ubuntu phone, I can't imagine that anybody else will have the

answers I need. Somehow, I don't think that BQ in Spain will know the answers either.

With the desktop Ubuntu, it's free, so you don't have a leg to stand on to complain about the lack of official support, but the phone costs money, and that's why I think that there should be some sort of official support. Perhaps Canonical can put an Ubuntu Phone page on their website with notifications about software updates, an FAQ, and perhaps even an opportunity to contact them if you have a question not answered in the FAQ. For instance, how will I know when the e-mail scope is ready? The phone may nudge me for an update, but will it tell me what's in that update? I'm happy to sign up with Canonical for notifications if such a thing was available. At present, I feel somewhat alone, struggling to solve the various issues myself (although Ronnie has been kind enough to help me out with a few things – thanks Ronnie!).

Perhaps all this is because the phone is so new? Maybe this is the norm for early adopters? It may well be, and I admit that I have never been an early adopter before, and after this experience I don't want to be one again! Next time, I will wait until the product is sorted before buying.

Reading back what I have written here is really rather negative, which is a shame, as I'm a fan of Ubuntu and really want it to succeed, but I feel that, if I don't speak up, Canonical won't know how/what to improve. A lot of mistakes have been made, in my opinion, but I'm confident that, once the phone is sorted, it'll be really good. I never took much notice of scopes on my desktop / laptop Ubuntu, but on the phone they really come to life. The phone itself is good quality and good value for money. It works quickly and smoothly. I'm looking forward to the day when all the various problems have been sorted and I can just get on with enjoying it and buy one for my non-technical wife, but for that it really has to

'just work'.

Linux Labs
Owncloud for fun and easy backups

By Charles McColm, chaslinux at gmail dot com

Entertainment systems like Kodi are great for storing videos, pictures, music and retro games, but what if you need to get a little more serious and back up important documents like your spouse's résumés or your growing collection of eBooks? You could use something like SAMBA and scripts/software to automate backing up the résumés to your SAMBA server, but how do you deal with mobile devices? Dropbox is another solution, but it stores your data on the cloud where it could end up as someone else's property. What you really want is your (Own)cloud.

Bad puns aside, Owncloud is personal client and server cloud software. Clients exist for Linux, Mac OS X, Windows, iOS, and Android devices. The server software is available for Linux and web hosts running either Windows server or Linux. On the Linux side, server packages are available for CentOS, Debian, Fedora, OpenSUSE, RedHat Enterprise Linux, SUSE Linux Enterprise, and Ubuntu.

To install owncloud on Ubuntu, we need to add the repository where owncloud is stored. Owncloud uses the OpenSUSE build service to build packages, so don't be surprised that the repository is hosted from their servers. Because instructions vary among versions of Ubuntu (and other flavours of Linux), you can find complete instructions here:

http://software.opensuse.org/download/package?project=isv:ownCloud:community&package=owncloud

In this example, I'll use the instructions for Ubuntu 14.04 (on an Xubuntu 14.04 system). I'm not sure why, but a lot of places seem to end their instructions with the repository key when it makes more sense to start with it. If you try adding the software first, you get scary messages about untrusted sources. Better to start with the key:

```
wget http://download.opensuse.org/repositories/isv:ownCloud:community/xUb

sudo apt-key add - < Release.key
```

The first line gets the key and stores it in the local directory; the second command imports the owncloud key. Now it's time to add the software repository:

```
sudo sh -c "echo 'deb http://download.opensuse.org/
repositories/isv:/ownCloud:/community/xUbuntu_14.04/
/' > > /etc/apt/sources.list.d/owncloud.list"
```

Note that the above command is all one line. Now we'll update the repositories and install the owncloud server:

```
sudo apt-get update

sudo apt-get install owncloud
```

If you're willing to forego the bit of automation the clients give, you can also use a modern web browser to drag and drop files to your owncloud server.

Apple iPad/iPhone owners should note that the iOS client is not free, but the current cost is far less than most mobile apps. You can always just use your browser as a client for uploading and downloading files from your owncloud, but you won't have the synchronization benefit of the client.

Linux users can simply download the owncloud client package:

```
sudo apt-get install owncloud-client
```

If you're hosting owncloud on a hosted service, you'll want to enable Secure Socket Layer (SSL) so your account credentials are not easily intercepted. I followed How To Create a SSL Certificate on Apache for Ubuntu 14.04 steps on Digital Ocean (note: I don't use Digital Ocean, but am thankful for their clear instruction):

https://www.digitalocean.com/community/tutorials/how-to-create-a-ssl-certificate-on-apache-for-ubuntu-14-04

If you're just synchronizing between home machines, then you might want to use just regular http. If you're mixing Windows and Linux machines, then you're going to need a way for the Windows machines to know the I.P. address of your server. The simplest way I know is to set up your Linux server with a static I.P. address, then add the I.P. address and name to the Windows hosts file. Rackspace has a good article on how to modify the Windows hosts file along with the location of the file for Windows 8/7 and Vista:

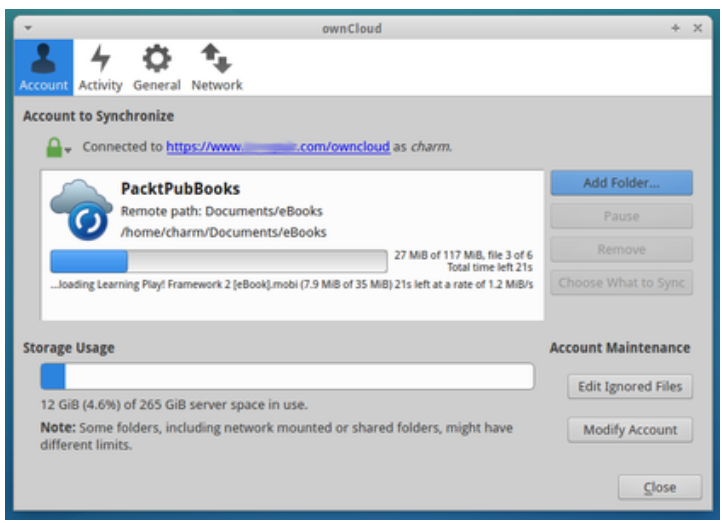
http://www.rackspace.com/knowledge_center/article/how-do-i-modify-my-hosts-file

To edit the hosts file, you're need to right-click on notepad and run as administrator, then open the hosts file. You can find the hosts file in Windows 8/7/Vista by navigating to C:32.

The format is &lt:ip address> hostname. So, if you call your Linux computer 'tank', and tank's IP address is 192.168.1.15, your Windows hosts file would contain:

192.168.1.15 tank

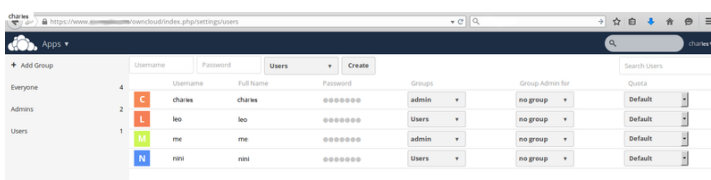
We're simply telling Windows where tank can be accessed. Of course, if you're a DNS wizard, and know all about DNS, or have DNS features built into your router, you might choose to give your server a DHCP reservation and do the DNS yourself. If you install owncloud in Ubuntu using the method above, you can connect via http or https<servername/ip>/owncloud. For example: http://192.168.1.15/owncloud/. The first user account will be your administrator, the account that can create other accounts.



You can synchronize any local folders with your owncloud by opening the client and clicking the Add Folder button on the right of the client. You'll be prompted to choose the local folder you want to synchronize, either enter it in or browse to it. You'll also have to enter an alias for the directory. You might be synchronizing a number of folders in different areas with similar content; using an alias can help keep them straight (e.g. UbuntueBooks, codingeBooks). Owncloud will then prompt you to choose the directory on the server to store

the synchronized content. If there are sub-folders, they are automatically selected for synchronization, but can be unchecked on the screen following the server directory selection. Don't forget synchronization goes both ways, so if you select something that already has a lot of content on your server, make sure you have space on your client and vice versa. Beside each file and folder (not to mention calendar) there's a share icon that lets you share data you want to share. The share can be password protected.

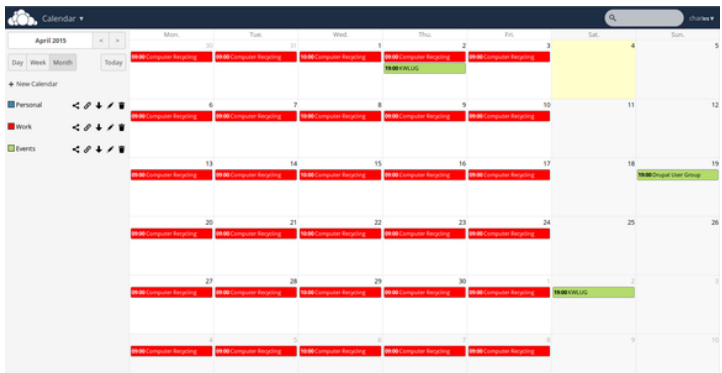
When you first set up owncloud, you'll want to log in through the web interface in order to add any extra users. My wife likes to have all our family photographs and she tends to store important documents on her computer (and I have a bad habit of changing hard drives/SSDs on her), so I've set up an account so she can synchronize what's important to her. The same for our son.



To access the Users menu, click your username in the web interface (provided you're the admin of course), and click the Users menu option. You can add a new user simply by entering the username and password in the username and password fields above the users, and clicking the Create button. If you're running an office with a lot of different users you might want to separate the users into different groups. Groups can be created either by clicking the Add Group button or clicking the Groups drop-down beside the password field. Owncloud can do a lot more than just synchronize documents and pictures. Add-ons exist so everyone can have their own calendar, bookmarks, and contacts. These features can really be handy if you're running a small/home business or just a busy person on

the go all the time.

I started using the calendar feature to keep track of user group meetings.



You can add a calendar for each type of event you have to help organize your time. All calendars appear on one calendar, but are differentiated by colour (in my example). I mark off work in red and user group meetings in green. This month, I haven't marked off any personal obligations (e.g. dentist). Calendars can be shown by Day/ Week/Month, and you can go forward and backward to check future and past events.

Owncloud isn't quite as easy as Dropbox to set up (if you have to set up the server side), but you can ensure that you own your own data ,and you're more likely to be the only set of eyes looking at your data.

*Charles McColm is the author of Instant XBMC, a short book on installing and configuring XBMCbuntu (a *buntu +XBMC distribution) - still relevant for Kodi, and the project manager of a not-for-profit computer reuse project. When not building PCs, removing malware, and encouraging people to use GNU/Linux, Charles works on reinventing his blog at http://www.charlesmccolm.com/.*

Q&A 96

By Gord Campbell

Q I bought 2 gaming computers running ubuntu 14.04 from an auction and they seem to be password protected. I have no clue how to bypass this as I didn't receive any login info for them.

A (Thanks to kerry_s and yancek in the Ubuntu Forums) I think you need to wipe them, which should have been done before they went to auction. This is very bad for the previous owner of those computers. I hope you're a good person.

It could also be bad for you, as the person who sold the computers may have left some back door or malware on them. I think either scenario is unlikely but also feel you would be better off with a new install as they could be cluttered up with all kinds of software and/or personal data.

Q I cannot seem to be able to get this 1 TB thumbdrive formatted and usable. It came with exFAT on it. I am trying to get Windows NTFS on it, or even FAT32. Last resort would be EXT3. Unfortunately, I keep coming up with errors no matter what one I try. I am using gparted to do this. Any suggestions to what I may be doing wrong?

A (Thanks to DuckHook in the Ubuntu forums) It's a scam; the actual capacity of the thumb drive is 8 GB.

Q How can I list all the hard drive partitions?

A Run this command: lsblk

Q When I try to use apt-get to install software, it doesn't work.

A Run this command: sudo apt-get update

Q Should I encrypt my Ubuntu installation?

A Encrypting your hard drive helps if someone steals your computer (or hard drive) and you have important personal information stored there. It has no other value. (**Comments welcome!**)

Q Can I block a destination based on IP address?

A (Thanks to gabriel40 in the Ubuntu forums) Yes, here's the command: iptables -A OUTPUT -j DROP -d w.x.y.z

Q When I connect my Ubuntu phone with USB cable, I can connect with adb, but I can't browse phone folders with nautilus.

A (Thanks to davidricq87 in the Ubuntu Forums) OK, libmtp is outdated. I used this ppa: https://launchpad.net/~phablet-team/+archive/ubuntu/tools and solved my problem.

Q When installing Ubuntu, if I don't check the box where it asks me if I want to install third party software (flash player, etc), do I then get a completely free system? Free as in free speech, not free beer. In other words, no proprietary anything.

A Yes.

Q Do I need a FireWall on my Ubuntu? If yes can you recommend one?

A (Thanks, in part, to OrangeCrate in the Ubuntu Forums) If you are connecting through a router, you probably do not need one.

Ubuntu comes with a firewall called iptables, and you can configure iptables with UFW (Uncomplicated Firewall). Instructions here: https://help.ubuntu.com/community/UFW

Top questions at Askubuntu

- How can I run a program and watch its activity using top?
  http://goo.gl/z5OcnR
- How to transfer the identity (SSH) of a machine to a new machine?
  http://goo.gl/xitkwj
- How to delete files listed in a text file
  http://goo.gl/7fLQfe
- How do I count text lines?
  http://goo.gl/Z1I15D
- Google Chrome PPA upgrade invalid signature
  http://goo.gl/wz3hP3
- Command Line - Move here [on hold]
  http://goo.gl/ZDMQme
- How to display the paths in $PATH separately

http://goo.gl/q4lURj

- Why I can't uncompress a .bz2 file using tar?
  http://goo.gl/KQDUIJ

- Total video time in a Directory
  http://goo.gl/ktJBQn

Tips and Techniques

Nice!

I began testing the alpha version of Xubuntu in early March. Fingers crossed, it has been rock solid! Unlike, for example, Ubuntu 14.04 LTS on my AMD Phenom II X2-based system.

It would be nice to credit Canonical with the improvement, but it's probably the 3.19 kernel. Years ago, Ubuntu 10.04 would run for a few hours, then lock right up. Ubuntu 10.10 ran fine, and so did 12.04. Since then, every version has reverted to the unacceptable behaviour of 10.04.

One of the new features involves CPU microcode, which, I must confess, is a higher technical level than I understand.

On March 9, Canonical switched from upstart to systemd, which is a huge "under the covers" shift. Smooth as silk!

For my testing, I created a bootable flash drive with the Xubuntu installer. Then I installed it (yes, a true installation) onto a 32 GB USB 3 flash drive plugged into a USB 2.0 port. It's a little slower than running from a hard drive, but not terribly so. I installed the Restricted Extras and an Additional Driver for my Nvidia video card.

Since then, I have installed most of the programs I normally use. Conky, VLC, Miro, KRDC, the weather app and LibreOffice all run fine. It appears that KRDC works even better than before, since I could send a Ctrl-Alt-Del to the remote computer.

Miro 6 was a big surprise: downloading multiple podcasts simultaneously takes much less CPU than before. (The heatsink on my CPU needs cleaning, so high CPU usage is a big problem.) However, Miro also crashed in a couple of interesting ways.

A couple of times, there has been a pop-up claiming that Chrome has "closed unexpectedly", but I saw no evidence that it had actually crashed.

To install f.lux I had to use the Trusty repository. That's typical of third-party software when you are testing a new release. But f.lux is not actually doing what it is supposed to do.

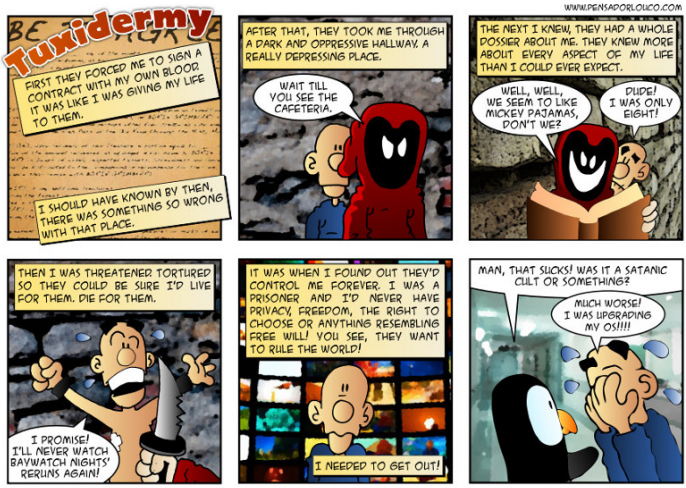
I've been keeping it up to date, which requires frequent large downloads; it's a "development release," after all.

At the office, we have Xubuntu on some low-spec (ex-XP) laptops, and there's no visual difference with 15.04, except perhaps the available wallpapers -- and that's a good thing! The application programs have been brought up to date. (LibreOffice!)

I also installed the 15.04 beta of Ubuntu Kylin on an old netbook with a large monitor attached. It was not a success. Once I pressed the mystery keys to get the Unity Launcher displayed, I was computing in slow motion. Kylin demands more horsepower. Xubuntu runs fine on the netbook.

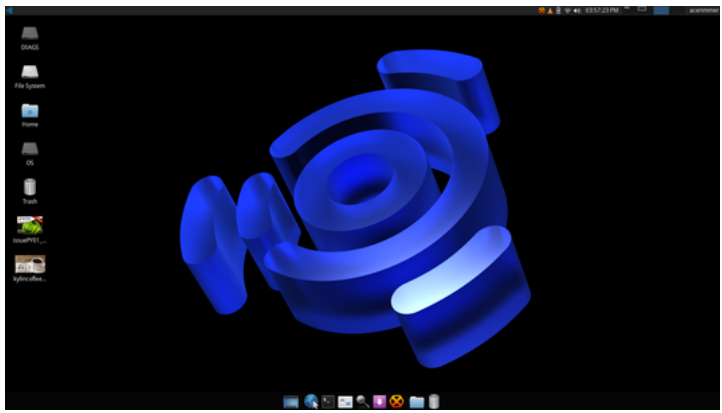
By the time you read this, Xubuntu 15.04 will be released. In my opinion, the only drawback is that you must

upgrade to 15.10 by late January of next year.


Tuxidermy
FIRST THEY FORCED ME TO SIGN A CONTRACT WITH MY OWN BLOOD. IT WAS LIKE I WAS GIVING MY LIFE TO THEM.
I SHOULD HAVE KNOWN BY THEN, THERE WAS SOMETHING SO WRONG WITH THAT PLACE.
AFTER THAT, THEY TOOK ME THROUGH A DARK AND OPPRESSIVE HALLWAY. A REALLY DEPRESSING PLACE.
WAIT TILL YOU SEE THE CAFETERIA.
THE NEXT I KNEW, THEY HAD A WHOLE DOSSIER ABOUT ME. THEY KNEW MORE ABOUT EVERY ASPECT OF MY LIFE THAN I COULD EVER EXPECT.
WELL, WELL, WE SEEM TO LIKE MICKEY PAJAMAS, DON'T WE?
DUDE! I WAS ONLY EIGHT!
THEN I WAS THREATENED TORTURED SO THEY COULD BE SURE I'D LIVE FOR THEM. DIE FOR THEM.
I PROMISE! I'LL NEVER WATCH BAYWATCH NIGHTS' RERUNS AGAIN!
IT WAS WHEN I FOUND OUT THEY'D CONTROL ME FOREVER. I WAS A PRISONER AND I'D NEVER HAVE PRIVACY, FREEDOM, THE RIGHT TO CHOOSE OR ANYTHING RESEMBLING FREE WILL! YOU SEE, THEY WANT TO RULE THE WORLD!
I NEEDED TO GET OUT!
MAN, THAT SUCKS! WAS IT A SATANIC CULT OR SOMETHING?
MUCH WORSE! I WAS UPGRADING MY OS!!!!

Review Dell Precision

by C. F. Howlett



Make: DELL
Model: Precision m3800 Developer EditionDELL
Category: laptop/workstationDELL
Distro: Ubuntustudio 14.04.2

When DELL announced Project Sputnik (http://www.dell.com/learn/us/en/555/campaigns/xps-linux-laptop?c=us&l=en&s=biz) I was intrigued. I tested an early XPS 13 Developer Edition but found it overpriced and underpowered. I was impatiently awaiting the release of the 2015 version when DELL unexpectedly announced the immediate availability of the Precision m3800 DE. A colleague visiting family in the USA agreed to hand-carry the device back to China thus avoiding the 18% import tax imposed on tech imports.

Ubuntu 14.04 is installed in OEM configuration. The OS appears to be standard Ubuntu supplemented by Nvidia GPU drivers. However, I prefer a dedicated /home and Ubuntustudio. I attempted to repartition with my 14.04.1 USB, but it would not boot. With 14.04.2, boot and

installation ran as smooth as warm butter.

Compared to my venerable 2009 DELL Inspiron 1545, this machine is a racehorse. I generally test system performance with high-resolution video and a .blender file. Full-performance settings on the Nvidia GPU did cause the fan to operate loudly and continuously, but Big Buck Bunny at 2k and 60 FPS played like a dream. I rendered a 1920 x 1080 scene in Blender in minutes, and the Half-Life: Lost Coast game on high-settings played at 48 FPS!

The grub and lightdm login screens boot to an unusable 3840 x 2160 resolution. I have corrected the grub screen and expect to fix the login screen. I've adopted the 1920 x 1080 resolution for daily use. DELL thoughtfully provides a utility to restore the laptop to its factory-fresh OEM configuration and an additional utility for conducting system diagnostics. Sadly, after installing Ubuntu Studio, grub failed to register these two partitions, although they remain on the device – I fully expect to recover them via grub or the command line in the next few days.

Physical design: very nice. The touchpad is a bit oversensitive, but that can be adjusted. I have yet to find a need for the touchscreen. The display is bright and beautiful. Back-lit keys work out of the box. Sound is great. The build quality is on par with Apple MacBook, and this thing is so LIGHT compared to my Inspiron! I've not yet tested bluetooth.

Thus far, color me impressed. This machine easily meets or beats MacBook specifications in many metrics. I now have an OEM supported Ubuntu laptop that delivers a direct retort to smug MacBook toting peers. In many respects, this is the portable multimedia workstation I had been waiting for.

Random coolness: the prominently displayed Ubuntu sticker!

Pro:

- lightweight
- powerful Nvidia Quadro K1100m graphics
- beautiful, high-resolution display
- touchscreen on-demand
- impressive performance

Con:

- unimpressive battery life
- noisy fan
- few apps currently exploit touch-screen capability
- standard re-installation of grub doesn't capture the custom DELL
- utility partitions.

SSH Authentication

by Kevin O'Brien

You can do this in several ways:

1. Password – You authenticate to the server by typing in your password. This is easy because you can generally remember your password, and it means you can easily login from any computer with that knowledge. This is still the most common authentication mechanism for SSH.
2. Public Key – This is much more secure. It involves the creation of a key pair, of course. It is possible to use a key pair generated by PGP or GPG in the most current versions (version 2.0.13 introduced support for this). But there is a long-established method using the Unix program ssh-keygen. This is very similar to generating a key pair as we discussed earlier. You run the program 'ssh-keygen', harvest some entropy, generate a passphrase to protect it, and so on.
3. Kerberos – This is done through the General Security Services API. This is a programming interface that is broader than just Kerberos—it is supposed to encompass several possibilities, and, of course, as an API it abstracts from the details. But the GSSAPI library included supports only Kerberos, so it is not yet as general as it might become.
4. Keyboard-Interactive – The server sends one or more prompts to the client to enter certain information. This is not compatible with all client software, however. But it would work in a terminal.

I won't go into detail on the last two of these, as I consider them very specialized. If you need to know more on either of them, a Google search would probably turn

up what you need. The most common methods are either entering a password, or using a public key. And, like so many things in security, there is a tradeoff between security and ease of use. Passwords are the simplest and easiest way to authenticate, and everyone knows how to use one. But it is also true that passwords can be compromised in a variety of ways. People may use a single password for everything, or they might use a password easily guessed. Or it might be written on a sticky note "hidden" under the keyboard. Or it might be given to someone else to use, particularly in a corporate environment where many people may need to access the same resources. Since the whole idea of using SSH is to increase the security, I don't like relying on passwords if there is an alternative. And increasingly, public key is that alternative.

Public Key Authentication

The starting point for this is generating the key pair. As we said previously in our tutorial on Symmetric vs. Asymmetric Encryption, there are several possible algorithms that can be used, with RSA still the most common. And the way this works is that you generate two keys, such that key1 will decrypt anything that key2 encrypted, and key2 will decrypt anything key1 encrypted. Arbitrarily, one of these is designated as the public key, and the other as the private key. For your algorithm, you generally choose either RSA, DSA, or ECDSA. RSA uses large prime numbers to get its keys, DSA (Digital Signature Algorithm) uses discrete logarithm mathematics, and ECDSA (Elliptic Curve DSA) use Elliptic Curve mathematics. All of these are examples of a one-way algorithm, which means that it uses a computation that is easy to do, but extremely difficult to do in reverse. Right now, it looks like RSA is more widely used, but DSA is slightly stronger, and ECDSA is fairly new but coming up fast because it is highly efficient. Since RSA is widely

used, it makes sense to go with RSA unless you have a strong reason not to.

Your next decision is the key length, and here the default should be 2048. 1024 is more than the current record for brute force cracking, but not that much more. If you have a powerful computer, go for 3072. If you want more information on the ssh-keygen command, try the man page. This should give you two files which will reside in the same ~/.ssh/ directory as your known-hosts file. One will be your ID name_rsa (assuming you used RSA), which will be your private key. The other will be your ID name_rsa.pub, which will be your public key.

If you are a Windows user, try downloading puttygen.exe, which works with PuTTY. Instructions can be found at https://kb.siteground.com/how_to_generate_an_ssh_key_on_windows_using_putty/. Your two files will be C:ID Name.sshID name_rsa (private key), and C:ID name.sshID name_rsa.pub

Once you have generated these keys, you need to add the public key to your ssh account on the server. How this happens may vary. On a more-or-less public system (for example, here is the GitHub explanation), they may let you add this through a Website where it is added to your account information. In a corporate environment, you may find that the IT department takes care of generating the key and adding it to the server. So you need to check with the server to see how they handle this.

If you have access to the server (eg, it is a server you administer), there is a file called $HOME/.ssh/authorized_keys that holds the public keys of users, one to a line (and the lines are long, of course). Since you have not yet uploaded your key, this login to the server will need to be authenticated by using a password, but once you have added it, you should be good in the future.

Simply copy the file with your public key, then cat the file to the authorized_keys file to add it. Note that if you are the admin on this server and have the rights needed to do this, make very certain that you set the permissions properly so that no one else who gets on your server can read the file. The idea is to be secure, after all.

Agents

In use, you would need to use your passphrase each time you tried to open an ssh session. While it is possible to create a public key without a passphrase, it is a very bad idea to do this. And a short, memorable passphrase it almost as bad. The passphrase needs to be long to do its job. I would recommend that you first store this in a secure vault like KeePassX (see the tutorial on Passwords, Entropy, and Good Password Practices for more on this http://www.zwilnik.com/?page_id=530). But if you do a lot of SSH sessions in a day, this does get old. Fortunately, there is a good, reasonably secure solution, called an SSH agent. This is a program that resides in memory and holds your decrypted private key. Every time you go to a site using SSH, it uses this key to generate a message which the server decrypts using your public key (which it has). And when you shut down your computer at the end of the day, it drops your key out of memory, so the next day when you boot up you need to enter your passphrase one more time. There are obvious security concerns here. If you don't lock your computer every time you walk away from it, your private key can be grabbed by anyone.

In any Unix-like system, the program ssh-agent should be installed. Many display managers will have hooks into the ssh-agent, and will find your key (xdm and kdm are two such) if it is in the default location. You will know this if, when you boot up, you get a pop-up window asking you for your passphrase. Ubuntu is a little different, so you can read the Ubuntu man page for this here. For windows

users, the PuTTY ssh agent is called Pageant. If you place a shortcut to it in the startup folder, it will run automatically every time you boot Windows.

Final Caution

There are some things to keep in mind here. First, just as with your PGP key for e-mail, which we discussed previously, if you lose your key you are in trouble. Backing up is important. If you don't back up your keys, you may find one day that you no longer have access to those remote systems. You might be able to get new access by deleting the old keys and getting new ones added, but if you log into a lot of sites that will be a royal pain. Also, what happens if a computer that has your keys on it is decommissioned, sold, or compromised in some way? How secure is your access now? One recommendation is that you don't use the same keys on different machines to help guard against this. It might seem like additional work to create key pairs on each machine separately, but if the point is security it just might be a good idea.

Ubuntu Games
Cities: Skylines

by Ronnie Tucker

It's been described as the game that SimCity 4 should have been. I can neither confirm nor deny that statement, as I've never played SimCity 4 since Maxis didn't have the decency to make a native Linux version. Colossal Order (a small Finnish team) did have the decency and foresight to make a Linux version of Cities: Skylines. And I'm so glad they did.

In The Beginning

On first starting the game you're given a choice of maps to choose from. They're different, but each contain some body of water (which you'll initially use to generate power), and mostly land to build upon.

You're given hints and tips along the way telling you to place roads, zones, electricity, water, and so on.

If you've ever played the original SimCity, or SimCity 2000, then you'll instinctively know what to do.

The Basics

Your map has two roads leading into it. These connect your future city to the highway which has vehicles coming/going to your city. You need to connect to these highways to allow your city to expand. Once you have some roads in place, you can place zones beside the roads. You have three choices (initially); residential, commercial, or industrial. Obviously you need to keep your (noisy, smelly) industrial zones away from your housing and put your commercial zones somewhere in the middle. This is where the detail in CS shines through. Bad planning means that your residents will complain about noise pollution from the factories. Yes, you need to take into account noise when building your town. Even small things such as wind turbines to initially give you electricity need to be kept away from residents. Details such as planting trees to block noise are here too.

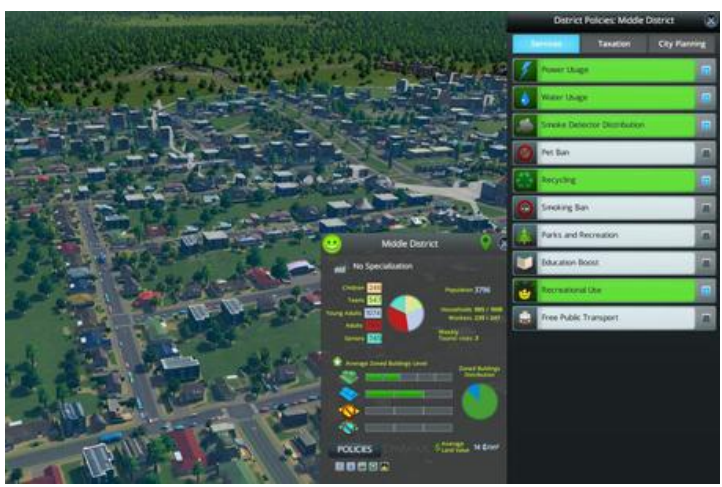
All zones need two basic commodities; electricity and water lines. You'll need to connect pumps to the water areas to pump clean water in. But watch out. You also need to connect pumps to dump your sewage. Dump your sewage upstream from your intake pump and, well, you can imagine the results. Wind turbines supply your power at the start, but, as you establish your city, you can use water/solar power or just good old fashioned coal.

Even placing roads has its good and bad points – placing the wrong road in a busy area means massive traffic jams and pollution. A planners life is not an easy one.

Stats

There's a ton of statistics and management in the game too. Top left of the screen gives you quick access to information regarding who has/hasn't got electricity, water, etc. The range the police/fire services cover. The aforementioned noise pollution. You name it, it's all here. Even down to how much electricity you're generating and how much your citizens are using. This is also shown when you click the building options at the bottom of the screen. Here you place roads, zones, buildings, etc.

Another really nice feature, and really well implemented, is the use of districts. You can choose an area of the city and give it a name. You can select it by using a box tool or by literally painting over the town. This means you can apply certain rules/policies to certain districts within the city.



Again, details such as forcing industrial districts to install fire alarms can reduce the strain on your fire department. Districts can also be given an industry. So, for example, you can select an area of nice countryside, give it a name,

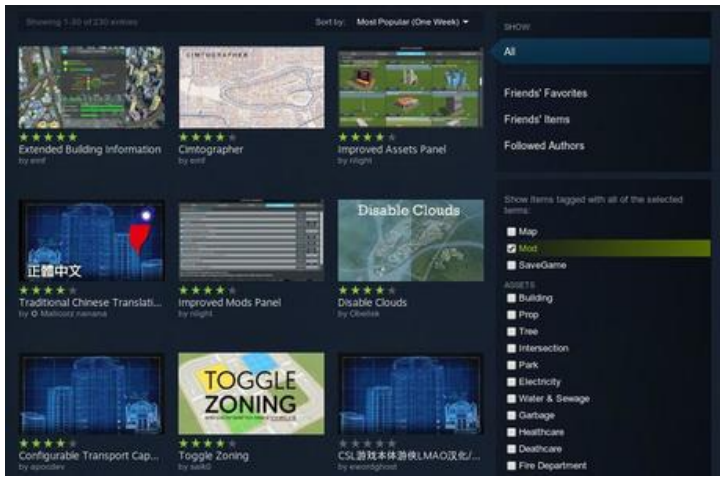
and use it only for farming. Oh, and everything – and I mean everything – has a name/title in CS, and you can change them all. There is much fun to be had renaming people, animals, and zones.

Speaking of people, everything in CS has a purpose. You can select any person and it will tell you who the person is, where they work, and/or where they are going. Same with vehicles.

I've only scratched the surface of the detail in CS. There's so much more such as buying adjacent land to expand your city, mining, and tweaking budgets.

Mods

As if all that goodness wasn't enough, the entire game is mod friendly. There are already hundreds of mods that will give new maps.



One awesome mod I have installed is one that gives you the ability to choose any citizen and look through their eyes. No, the graphics won't look hugely impressive (as you were really never meant to get in that close), but it's great that the developers allow such mods.

Conclusion

If you hadn't already guessed: I love CS.

CS uses the Unity engine, and it really is nice. Graphics settings are highly adjustable to allow play even on lower-end machines. My machine has a lowly GeForce GT640, and it runs the game great even with tilt-shift, medium shadows, medium detail, and filtering on. My city isn't a sprawling metropolis, but there's a lot going on in there for my little graphics card.

Pointing out every feature of CS would fill a large book, so if you loved SimCity 2000, or hated SimCity 4 (for various reasons), then give Cities: Skylines a look. Cities: Skylines is available through Steam for USD $29.99.

Rating: 5/5

System Requirements

Minimum:

- Operating System: Ubuntu 14.04 64-bit, Ubuntu 14.10 64-bit, Ubuntu 12.04 64-bit, Ubuntu 14.04

(64-bit)
- Processor: Intel Core 2 Duo 3.0GHz / AMD Athlon 64 X2 6400+ 3.2GHz
- RAM: 4GB
- Graphics Card: nVIDIA GeForce GTX 260, 512 MB / ATI Radeon HD 5670 (Does not support Intel Integrated Graphics Card), 512 MB
- Hard Drive: 4GB

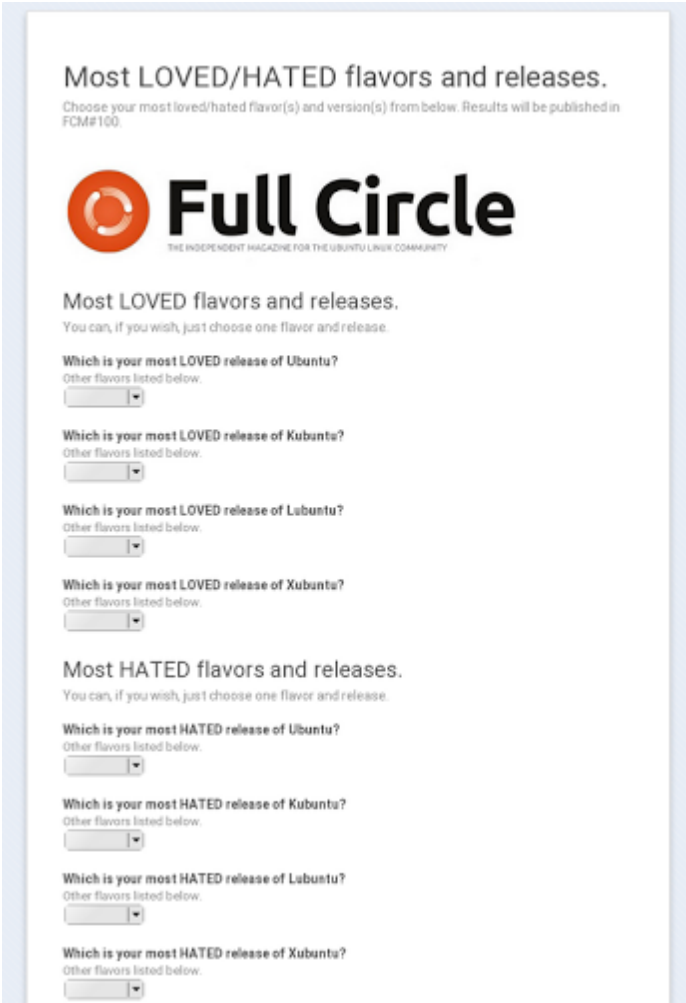
Recommended:

- Operating System: Ubuntu 14.04 64-bit, Ubuntu 14.10 64-bit, Ubuntu 12.04 64-bit, Ubuntu 14.04 (64-bit)
- Processor: Intel Core i5-3470 3.20GHz / AMD FX-6300 3.5Ghz RAM: 6GB
- Graphics Card: nVIDIA GeForce GTX 660 (Does not support Intel Integrated Graphics Card), 2GB / AMD Radeon HD 7870, 2 GB
- Hard Drive: 4GB

My Desktop

Again there is no content for these pages

On the other hand we do have a survey at http://goo.gl/DPt2q0



Most LOVED/HATED flavors and releases.

Choose your most loved/hated flavor(s) and version(s) from below. Results will be published in FCM#100.

Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

Most LOVED flavors and releases.

You can, if you wish, just choose one flavor and release.

Which is your most LOVED release of Ubuntu?

Other flavors listed below.

Which is your most LOVED release of Kubuntu?

Other flavors listed below.

Which is your most LOVED release of Lubuntu?

Other flavors listed below.

Which is your most LOVED release of Xubuntu?

Other flavors listed below.

Most HATED flavors and releases.

You can, if you wish, just choose one flavor and release.

Which is your most HATED release of Ubuntu?

Other flavors listed below.

Which is your most HATED release of Kubuntu?

Other flavors listed below.

Which is your most HATED release of Lubuntu?

Other flavors listed below.

Which is your most HATED release of Xubuntu?

Other flavors listed below.

FLAT
20%
Discount
[PACKT]
PUBLISHING
Learning AndEngine
Martin Varga
Offer Valid till - 21st November 2014
Use Discount Code - gUdb4pzLI
http://bit.ly/1rG1qDE

Contribute to Full Circle Magazine
FULL CIRCLE NEEDS YOU!

A magazine isn't a magazine without articles – and Full Circle is no exception. We need your opinions, desktops, stories, how-to's, reviews, and anything else you want to tell your fellow *buntu users. Send your articles to: articles@fullcirclemagazine.org.

We are always looking out for new articles to include in Full Circle. For help and advice, please see the Official Full Circle Style Guide: http://url.fullcirclemagazine.org/75d471

Send your comments or Linux experiences to: letters@fullcirclemagazine.org
Hardware/software reviews should be sent to: reviews@fullcirclemagazine.org
Questions for Q&A should go to: questions@fullcirclemagazine.org
Desktop screens should be emailed to: misc@fullcirclemagazine.org
... or you can visit our forum via: www.fullcirclemagazine.org

Full Circle Team:
Editor - Ronnie Tucker - ronnie@fullcirclemagazine.org
Webmaster - Rob Kerfia - admin@fullcirclemagazine.org
Podcast - Les Pounder & Co. - podcast@fullcirclemagazine.org

Editing & Proofreading:
Mike Kennedy
Lucas Westermann
Gord Campbell

Robert Orsino
Josh Hertel
Bert Jerred

Our thanks go out to Canonical, to the many translation teams around the world, and to Thorsten Wilms for the current Full Circle logo.

Getting Full Circle Magazine:

EPUB Format - Most editions of Full Circle have a link to the epub file on the downloads page. If you have any problems with the epub file, you can drop an email to: mobile@fullcirclemagazine.org

Google Currents - Install the Google Currents app on your Android/Apple devices, search for 'full circle' (within the app), and you'll be able to add issues 55+. Or, you can click the links on the FCM download pages.

Ubuntu Software Centre - You can get FCM via the Ubuntu Software Centre: https://apps.ubuntu.com/cat/. Search for 'full circle', choose an issue, and click the download button.

Issuu - You can read Full Circle online via Issuu: http://issuu.com/fullcirclemagazine. Please share and rate FCM as it helps to spread the word about FCM and Ubuntu Linux.